



# Gestion de la mobilité dans les réseaux Ad Hoc par anticipation des métriques de routage

Sabrine Naimi

## ► To cite this version:

Sabrine Naimi. Gestion de la mobilité dans les réseaux Ad Hoc par anticipation des métriques de routage. Physique des Hautes Energies - Réseau [hep-lat]. Université Paris Sud - Paris XI; École nationale d'ingénieurs de Tunis (Tunisie), 2015. Français. NNT : 2015PA112145 . tel-01208152

**HAL Id: tel-01208152**

**<https://theses.hal.science/tel-01208152>**

Submitted on 2 Oct 2015

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# UNIVERSITÉ PARIS-SUD

ÉCOLE DOCTORALE STITS  
LABORATOIRE DES SIGNAUX ET SYSTÈMES (L2S)

DISCIPLINE : PHYSIQUE

## THÈSE DE DOCTORAT

Soutenue le 22 juillet 2015 par

**Sabrine NAIMI**

## **Gestion de la mobilité dans les réseaux Ad Hoc par anticipation des métriques de routage**

<b>Directeur de thèse :</b>	Mme Véronique VÈQUE	Professeur (Université Paris-Sud)
<b>Directeur de thèse :</b>	Mr Ridha BOUALLEGUE	Professeur (Université de Carthage)
<b>Co-directeur de thèse :</b>	Mr Anthony BUSSON	Professeur (Université Lyon 1)

**Composition du jury :**

Président du jury :	Mr Anis LAOUITI	Professeur (Télécom SudParis)
Rapporteurs :	Mme Béatrice PAILLASSA	Professeur (Université de Toulouse)
	Mme Kaouther SETHOM	Maîtres de conférences (Université de Carthage)
Examineurs :	Mme Anne FLADENMULLER	Maîtres de conférences (Université Pierre et Marie Curie)



UNIVERSITÉ PARIS-SUD

## *Résumé*

LABORATOIRE DES SIGNAUX ET SYSTEMES

Supelec

Docteur en Philosophie

### **Gestion de la mobilité dans les réseaux Ad Hoc par anticipation des métriques de routage**

par Sabrina NAIMI

Avec le succès des communications sans fil, il devient possible d'accéder au réseau partout et à tout moment sans avoir recours à connecter physiquement les appareils communicants à une infrastructure. Les noeuds (ordinateurs portables, smartphones, etc.) peuvent analyser les différents canaux radio afin de pouvoir s'associer à un réseau sans fil disponible (station de base, point d'accès, etc.). Un avantage indéniable de ses technologies sans fil est la possibilité d'être mobile tout en restant connecté. Cependant, la mobilité est une tâche difficile à gérer car elle doit être abordée à différentes couches pour être transparente aux utilisateurs. Dans les MANET (Mobile Ad hoc Network), les protocoles de routage utilisent des métriques pour sélectionner les meilleures routes. Les métriques peuvent refléter la qualité de la liaison sans fil et aider à gérer la mobilité. Mais, un retard important entre l'estimation des métriques et leur inclusion dans le processus de routage rend cette approche inefficace.

Les travaux de cette thèse s'intéressent à la proposition de nouvelles méthodes de calcul des métriques de routage pour gérer le problème de la mobilité dans les réseaux ad hoc. Les nouvelles métriques doivent refléter la qualité du lien et être sensibles à la mobilité en même temps. Nous considérons les métriques classiques, en particulier ETX (Expected Transmission Count) et ETT (Expected Transmission Time). Nous introduisons de nouvelles méthodes pour anticiper les valeurs de ces métriques en utilisant des algorithmes de prédiction. Nous utilisons une approche Cross Layer, qui permet l'utilisation conjointe de l'information à partir des couches 1, 2 et 3. La validation de nouvelles méthodes de calcul des métriques de routage nécessite une évaluation au travers d'un véritable banc d'essai. Nous avons donc également mis en oeuvre les nouvelles métriques de routage dans un testbed afin d'évaluer et de comparer leurs performances avec les métriques classiques.

***Dédicace***

*Je dédie ce travail à ma chère fille MARAM (MIMI)*

*J'espère que tu seras fière de moi*

*Je t'aime énormément ma MIMI ♡*

## *Remerciements*

Je tiens à adresser en premier lieu mes plus chaleureux remerciements à ma directrice de thèse, Véronique VEQUE, pour m’avoir confié ce travail de recherche, ainsi que pour son aide et ses précieux conseils au cours de ces années. Je suis très honorée de l’avoir eu pour directrice.

Je tiens à exprimer mes plus vifs remerciements à mon directeur de thèse, Ridha BOUAL-LEGUE, pour m’avoir appris à être plus autonome tout au long de ce travail. Je suis ravie d’avoir travaillé en sa compagnie au cours de cette thèse.

Je souhaiterais exprimer ma gratitude à mon co-encadrant, Anthony BUSSON, pour la confiance qu’il m’a témoigné du début à la fin du travail, pour sa grande disponibilité à comprendre et à communiquer, sans oublier ses précieuses intuitions. Je suis ravie d’avoir travaillé en sa compagnie car, outre son appui scientifique, il a toujours été là pour me soutenir et me conseiller au cours de l’élaboration de cette thèse. Enfin, j’ai été extrêmement sensible à ses qualités humaines d’écoute et de compréhension tout au long de ce travail doctoral. Sincèrement, grâce à lui j’ai pu apprendre beaucoup de choses dont certaines fort utiles pour mes travaux académiques bien sûr, mais aussi des choses importantes pour mon développement personnel.

Je voudrais remercier vivement Madame Béatrice PAILLASSA, Professeur à l’Université de Toulouse et Madame Kaouther SETHOM, Maîtres de conférences de l’Université de Tunis Carthage pour avoir consacré du temps à la lecture de ce document en tant que rapporteurs.

J’associe à ces remerciements Madame Anne FLADENMULLER, Maîtres de conférences de l’Université Pierre et Marie Curie, et Monsieur Anis LAOUITI, Professeur de Télécom SudParis, pour avoir accepté d’examiner mon travail.

Au cours de ces années et dans le cadre de ma thèse en cotutelle, j’ai eu la chance d’être membre du pôle Telecoms et Réseaux du laboratoire des Signaux et Systèmes (LSS) de Supélec et aussi membre du laboratoire des Systèmes de Communications (SYS’COM) de l’École Nationale d’Ingénieurs de Tunis (ENIT). J’ai pu partager des bons moments avec de nombreuses personnes attachantes. Je pense notamment de Supélec à : Assia, Boutheina, Sarra, Iyad, Lynda, Safaa, Mohamed, Mahmoud, Zeina, Ali et de l’ENIT à : Hafewa, Ihsen, Marwa, Walid, Zied, Mohamed Bedoui, Mohamed, Meriem, Mourad, Houssein, Dorsaf, Zeineb, etc.

Je témoigne toute ma reconnaissance amicale à Racha Sellami et Ines Hassoumi pour leurs soutiens et leurs bonnes humeurs pendant mes stages à Paris. Elles ont rendu mes séjours plus agréables à vivre en leurs compagnies.

Ces remerciements ne peuvent s’achever, sans une pensée pour ma chère mère. Ses sacrifices, sa présence et ses encouragements sont pour moi les piliers fondateurs de ce que je suis et de ce que je fais. Un très grand merci aussi à mon cher papa pour son amour, ses encouragements et

ses sacrifices. Ma grande gratitude va également à ma chère tata Zohra, pour son soutien infini pendant tous mes états de morale. Je vous aime énormément maman, papa et tata. Que Dieu vous protège et vous garde pour nous. Vient le moment de remercier mon frère Dali et mes deux soeurs Abir et Rania. Je les remercie énormément pour leur soutien sans failles et tout ce qu'ils ont pu m'apporter pour franchir les obstacles les plus difficiles. Je vous aime. Je remercie aussi mes chères tatas Saliha et Jamila et mon oncle Abdel Waheb pour leurs soutiens et leurs confiances en moi. Ils ont toujours cru en mon potentiel et n'ont jamais cessé de m'encourager. Mes chères cousines Islem, Omayma, Asma, mes nièces Malak et Yasmine, et mon neveu Yasmine, merci pour votre présence dans ma vie. Je remercie toute ma famille sans exception.

J'ai choisi de les laisser à la fin, mes gros bébés, mes trésors : mon cher époux Zied, mes deux filles Mimi et Kinzy. Je te remercie Zaydouny pour avoir subi mes moments de stress d'angoisse et de fatigue durant ces années de thèse. Il a su m'écouter, supporter ma mauvaise humeur et mon caractère insupportable. Je le remercie pour son soutien quotidien indéfectible et son enthousiasme contagieux à l'égard de mes travaux comme de la vie en général. Je te remercie chéri pour ta patience, pour ton soutien, pour ta tendresse etc. tous simplement pour ton amour qui ne cesse pas de grandir. Je remercie mes deux anges Mimi et Kinzy : je suis désolée de ne pas être toujours disponible. Vous êtes mon «beau soleil» qui m'a apporté de la lumière dans ma vie pas très gaie à cette phase cauchemardesque qu'est la fin de thèse. Notre famille a grandi en même temps que mon projet scientifique, le premier servant de socle solide à l'épanouissement du second.

Je tiens à remercier aussi ma belle-mère Fatim Rebhi, mes belles soeurs Afra et Sondess, et Mohamed Naimi qui m'ont aidé et soutenu au cours de cette thèse.

Enfin, je présente mes excuses aux personnes que j'ai peut-être oubliées de citer, et qui pourtant ont été présentes lors de cette thèse, et j'adresse encore une fois mes remerciements à tous ceux qui, de loin ou de près, ont contribué à la réussite de ce présent travail.

Une pensée pour terminer ces remerciements pour toi qui n'a pas vu l'aboutissement de mon travail mais je sais que tu en aurais été très fier de ta petite fille !!!

# Table des matières

<b>Résumé</b>	<b>ii</b>
<b>Remerciements</b>	<b>iv</b>
<b>Liste des Figures</b>	<b>ix</b>
<b>Liste des tableaux</b>	<b>xi</b>
<b>Introduction générale</b>	<b>1</b>
0.1 Contexte de la thèse . . . . .	1
0.2 Organisation du mémoire . . . . .	3
0.3 Liste des publications . . . . .	4
<b>1 Les réseaux mobiles ad hoc</b>	<b>5</b>
1.1 Les réseaux mobiles ad hoc . . . . .	7
1.1.1 Définition des réseaux mobiles ad hoc . . . . .	7
1.1.2 Applications des réseaux mobiles ad hoc . . . . .	8
1.2 Mobilité dans les réseaux ad hoc . . . . .	9
1.2.1 Définition . . . . .	9
1.2.2 Modèles de mobilité pour les réseaux ad hoc . . . . .	11
1.3 Protocoles de routage dans les réseaux ad hoc . . . . .	15
1.3.1 Problème du routage dans les réseaux ad hoc . . . . .	15
1.3.2 Classification des protocoles de routage . . . . .	15
1.4 Le protocole de routage OLSR . . . . .	23
1.4.1 Découverte des voisins . . . . .	24
1.4.2 Découverte de la topologie . . . . .	26
1.4.3 Choix du protocole de routage OLSR . . . . .	27
1.5 Conclusion . . . . .	28
<b>2 Classification des métriques dans les réseaux ad hoc</b>	<b>29</b>
2.1 Métriques de routage . . . . .	31
2.2 Métrique basée sur la topologie du réseau . . . . .	32
2.2.1 La métrique Hop Count . . . . .	33
2.3 Métriques basées sur la mesure de la puissance du signal . . . . .	34
2.4 Métriques de la qualité de lien . . . . .	35



2.4.1	La métrique ETX . . . . .	35
2.4.2	La métrique ETT . . . . .	37
2.4.3	La métrique MTM . . . . .	38
2.5	Métriques de charge du trafic . . . . .	39
2.5.1	La métrique EAB . . . . .	39
2.5.2	La métrique MF-Transmission Failure . . . . .	40
2.5.3	La métrique C2WB . . . . .	41
2.5.4	La métrique MIND . . . . .	42
2.6	Les métriques muti-canaux . . . . .	43
2.6.1	La métrique WCETT . . . . .	43
2.6.2	La métrique MIC . . . . .	46
2.6.3	La métrique MCR . . . . .	47
2.6.4	La métrique iAWARE . . . . .	47
2.6.5	Les métriques mETX & ENT . . . . .	48
2.6.6	La métrique DBETX . . . . .	49
2.6.7	Les métriques BATD & iBATD . . . . .	50
2.7	Métriques sensibles à la mobilité . . . . .	52
2.7.1	Les métriques Link Associativity Ticks & Path Aaverage degree of as- sociation stability . . . . .	52
2.7.2	Les métriques Link Affinity & Path Stability . . . . .	53
2.7.3	Les métriques Link Availibity & Path Availability . . . . .	54
2.7.4	La métrique LD . . . . .	55
2.7.5	La métrique Link Change Rate, Link State Changes $\lambda_{lc}$ . . . . .	57
2.7.6	La métrique Link Stability $L_s$ . . . . .	57
2.8	Taxonomie des métriques de routage . . . . .	58
2.9	Choix des métriques . . . . .	58
2.10	Conclusion . . . . .	59
<b>3</b>	<b>Implémentation &amp; Évaluation des métriques ETX, ETT et LD</b>	<b>66</b>
3.1	Calcul & implémentation des métriques ETX & ETT . . . . .	67
3.1.1	ETX . . . . .	67
3.1.2	ETT . . . . .	71
3.2	Calcul & implémentation de la métrique LD . . . . .	73
3.3	Topologies de référence . . . . .	75
3.3.1	Topologie chaîne . . . . .	75
3.3.2	Topologie mesh . . . . .	76
3.3.3	Topologie ad hoc . . . . .	76
3.4	Évaluation des performances . . . . .	77
3.4.1	Configuration de la simulation . . . . .	77
3.4.2	Résultats pour la chaîne de noeud . . . . .	78
3.4.3	Résultats pour le réseau mesh . . . . .	81
3.4.4	Résultats pour la topologie aléatoire . . . . .	84
3.5	Conclusion . . . . .	87
<b>4</b>	<b>Algorithmes d'anticipation des métriques de routage</b>	<b>89</b>
4.1	Algorithmes d'anticipation des métriques de routage . . . . .	90
4.1.1	Anticipation basée sur le développement de Taylor . . . . .	91

4.1.2	Anticipation basée sur la prédiction de la puissance du signal reçu . . .	94
4.2	Évaluation des Performances . . . . .	100
4.2.1	Topologie chaîne . . . . .	102
4.2.2	Topologie mesh . . . . .	109
4.2.3	Topologie aléatoire . . . . .	113
4.3	Conclusion . . . . .	120
<b>5</b>	<b>Mise en Oeuvre du TESTBED</b>	<b>122</b>
5.1	Testbed pour les réseaux ad hoc . . . . .	123
5.1.1	Testbed . . . . .	123
5.2	Description de notre testbed . . . . .	124
5.2.1	Paramètres du testbed . . . . .	124
5.2.2	Description du scénario . . . . .	130
5.3	Résultats et discussion . . . . .	131
5.4	Conclusion . . . . .	136
	<b>Conclusion</b>	<b>137</b>
<b>A</b>	<b>ANNEXE A</b>	<b>139</b>
A.1	Configuration en mode ad hoc en utilisant Network Manager . . . . .	139
A.2	Installation d'OLSRD . . . . .	140
A.3	Iperf . . . . .	141
	<b>Bibliographie</b>	<b>143</b>

# Table des figures

1.1	Changement de la topologie à cause de la mobilité . . . . .	7
1.2	Exemples d'applications MANET . . . . .	9
1.3	Délai généré à cause de la mobilité . . . . .	10
1.4	Trajectoire d'un mobile générée par Random Walk . . . . .	12
1.5	Trajectoire d'un mobile générée par Random WayPoint . . . . .	13
1.6	Trajectoire d'un mobile générée par Random Direction . . . . .	14
1.7	Classification des protocoles de routage MANET . . . . .	23
1.8	Détection du voisinage par l'échange de messages HELLO . . . . .	25
1.9	La base Link_Tuple d'un nœud dans OLSR . . . . .	25
1.10	Comparaison entre une inondation simple et l'utilisation des MPR . . . . .	26
1.11	Exemple de routage dans OLSR . . . . .	27
2.1	Exemple de topologie de réseaux ad hoc . . . . .	33
2.2	Métrique ETX dans un exemple de topologie de réseau ad hoc . . . . .	36
2.3	Métrique ETT dans un exemple de topologie de réseau ad hoc . . . . .	36
2.4	Exemple de la métrique WCETT . . . . .	44
2.5	Exemple de calcul de la métrique LD . . . . .	56
3.1	Exemple de calcul de la métrique LD . . . . .	74
3.2	Scénario 1 : chaîne de nœud . . . . .	76
3.3	Scénario 2 : topologie mesh . . . . .	76
3.4	Scénario 3 : topologie aléatoire . . . . .	77
3.5	PDR en fonction de la vitesse pour ETX (Chaîne de noeuds) . . . . .	79
3.6	Comparaison du PDR en fonction de la vitesse (Chaîne de noeuds) . . . . .	80
3.7	Capacité en fonction du débit de la source (Chaîne de noeuds) . . . . .	81
3.8	Délai de bout-en-bout en fonction de la vitesse (Chaîne de noeuds) . . . . .	81
3.9	PDR en fonction de la vitesse pour ETX (Mesh) . . . . .	82
3.10	Comparaison du PDR en fonction de la vitesse (Mesh) . . . . .	83
3.11	Capacité en fonction du débit de la source (Mesh) . . . . .	83
3.12	Délai de bout-en-bout en fonction de la vitesse (Mesh) . . . . .	84
3.13	PDR en fonction de la vitesse pour ETX (Topologie ad hoc) . . . . .	85
3.14	Comparaison du PDR en fonction de la vitesse (Topologie ad hoc) . . . . .	85
3.15	Capacité en fonction du débit de la source (Topologie ad hoc) . . . . .	86
3.16	Délai de bout-en-bout en fonction de la vitesse (Topologie ad hoc) . . . . .	87
4.1	FER en fonction de SNR . . . . .	94
4.2	ETX et ETX_ANT en fonction de SNR . . . . .	95
4.3	Comparaison entre la puissance du signal réelle mesurée et prédite . . . . .	99

4.4	PDR en fonction de la vitesse avec l'algorithme 1	103
4.5	PDR en fonction de la vitesse avec l'algorithme 2	103
4.6	Algorithme 1 : PDR en fonction de la vitesse avec du fading	104
4.7	Algorithme 2 : PDR en fonction de la vitesse avec du fading	104
4.8	Comparaison des deux algorithmes (Chaîne de noeuds)	105
4.9	Algorithme 1 : capacité de toutes les métriques (Chaîne de noeuds)	106
4.10	Algorithme 2 : capacité de toutes les métriques (Chaîne de noeuds)	106
4.11	Capacité des métriques anticipées (Chaîne de noeuds)	107
4.12	Algorithme 1 : Délai de bout-en-bout (Chaîne de noeuds)	107
4.13	Algorithme 2 : Délai de bout-en-bout (Chaîne de noeuds)	108
4.14	Délai de bout-en-bout pour les métriques anticipées	108
4.15	PDR en fonction de la vitesse avec l'algorithme 1 (Mesh)	109
4.16	PDR en fonction de la vitesse avec l'algorithme 2 (Mesh)	110
4.17	Algorithme 1 : PDR en fonction de la vitesse avec du fading	110
4.18	Algorithme 2 : PDR en fonction de la vitesse avec du fading	111
4.19	Comparaison des deux algorithmes (Mesh)	111
4.20	Algorithme 1 : capacité de toutes les métriques (Mesh)	113
4.21	Algorithme 2 : capacité de toutes les métriques (Mesh)	113
4.22	Capacité des métriques anticipées (Mesh)	114
4.23	Algorithme 1 : Délai de bout-en-bout (Mesh)	114
4.24	Algorithme 2 : Délai de bout-en-bout (Mesh)	114
4.25	Délai de bout-en-bout des métriques anticipées (Mesh)	115
4.26	PDR en fonction de la vitesse avec l'algorithme 1	115
4.27	PDR en fonction de la vitesse avec l'algorithme 2	116
4.28	Algorithme 1 : PDR en fonction de la vitesse avec du fading	116
4.29	Algorithme 2 : PDR en fonction de la vitesse avec du fading	117
4.30	Comparaison des deux algorithmes (Topologie ad hoc)	117
4.31	Algorithme 1 : capacité de toutes les métriques (Topologie ad hoc)	118
4.32	Algorithme 2 : capacité de toutes les métriques (Topologie ad hoc)	118
4.33	Capacité des métriques anticipées (Topologie ad hoc)	119
4.34	Algorithme 1 : Délai de bout-en-bout (Topologie ad hoc)	119
4.35	Algorithme 2 : Délai de bout-en-bout (Topologie ad hoc)	120
4.36	Délai de bout-en-bout des métriques anticipées (Topologie ad hoc)	120
5.1	Configuration en mode ad hoc par script Shell	125
5.2	Configuration d'OLSRD à partir du fichier <i>olsrd.conf</i>	126
5.3	Ping en broadcast et en unicast pour le Mapping FER < — > Signal	128
5.4	Topologie du testbed	130
5.5	Voisinage des noeuds du testbed	131
5.6	Packet loss en fonction des Hellos pour Hop Count, ETX et ETX_ANT (TC = 2s)	133
5.7	Packet loss en fonction des Hellos pour Hop Count, ETX et ETX_ANT (TC = 5s)	134
5.8	Affichage de route par la commande "traceroute"	134
5.9	Sélection de route pour chaque métrique	135
A.1	Configuration en mode ad hoc en utilisant Network Manager	139
A.2	Configuration iperf entre client/serveur	141
A.3	Commandes iperf pour la mesure de performance entre client/serveur	142

# Liste des tableaux

1.1	Critères adaptés pour les protocoles de routage MANET . . . . .	22
2.1	Objectifs d'optimisation d'une métrique de routage . . . . .	31
2.2	Table de routage du nœud A avec la métrique Hop Count . . . . .	34
2.3	Coût des chemins possibles avec la métrique ETX . . . . .	37
2.4	Calcul de la métrique WCETT . . . . .	45
2.5	Taxonomie des métriques : Objectifs d'optimisation . . . . .	60
2.6	Taxonomie des métriques : Objectifs d'optimisation (suite) . . . . .	61
2.7	Taxonomie des métriques : Propriétés . . . . .	62
2.8	Taxonomie des métriques : Propriétés (suite) . . . . .	63
2.9	Taxonomie des métriques : faisabilité . . . . .	64
2.10	Taxonomie des métriques : faisabilité (suite) . . . . .	65
3.1	Paramètres de simulation . . . . .	77
4.1	Paramètres de simulation . . . . .	101
4.2	Table computing pour le nœud destination 10.1.0.15 à $T = 2061$ millisecondes avec ETX et ETX_ANT . . . . .	112
5.1	Cartes Wi-Fi internes utilisées dans le testbed . . . . .	124
5.2	Correspondance FER & Signal en broadcast . . . . .	129
5.3	Correspondance FER & Signal en unicast . . . . .	129
5.4	Correspondance FER & Signal . . . . .	129
5.5	Paramètres du testbed . . . . .	131
5.6	Packet loss des différentes métriques (marche lente) . . . . .	132
5.7	Packet loss des différentes métriques (marche rapide) . . . . .	132

# Abréviations

<b>ABR</b> Associativity Based Routing .....	20
<b>ACK</b> ACKnowledgement packet .....	35
<b>AODV</b> Ad Hoc On-Demand Distance Vector .....	19
<b>BATD</b> Bottleneck Aware Routing .....	50
<b>BER</b> Bit Error Rate .....	49
<b>BF</b> Bellman Ford .....	16
<b>BQ</b> Broadcast Query .....	21
<b>CBRP</b> Cluster Based Routing Protocol .....	20
<b>CGSR</b> Clusterhead Gateway Switch Routing .....	17
<b>CSC</b> Channel Switching Cost .....	46
<b>CV</b> Constant Velocity .....	14
<b>C2WB</b> Channel Utilization & Contention Window Based metric .....	41
<b>DAG</b> Directed Acyclic Graph .....	20
<b>DBETX</b> Distribution Based Expected Transmission Count .....	49
<b>DSDV</b> Dynamic destination Sequenced Distance Vector .....	17
<b>DREAM</b> Distance Routing Effect Algorithm for Mobllity .....	18
<b>DSR</b> Dynamic Source Routing .....	19

<b>EAB</b> Expected Available & Bandwidth .....	39
<b>ENT</b> Effective Number of Transmission .....	49
<b>ETX</b> Expected Transmission Count .....	35
<b>ETT</b> Expected Transmission Time .....	37
<b>FSR</b> Fisheye State Routing .....	17
<b>GSM</b> Global System for Mobile communication .....	6
<b>GSR</b> Global State Routing .....	17
<b>HSR</b> Hierarchical State Routing .....	18
<b>iAWARE</b> interference AWARE .....	47
<b>iBATD</b> improved Bottleneck Aware Transmission Delay .....	51
<b>IETF</b> Internet Engineering Task Force .....	15
<b>iETT</b> improved Expected Transmission Time .....	51
<b>IRU</b> Interference-aware Resource Usage .....	46
<b>LD</b> Link Duration .....	55
<b>LTE</b> Long Term Evolution .....	6
<b>MAC</b> Medium Access Control .....	17
<b>MANET</b> Mobile Ad hoc NETWORK .....	7
<b>MCR</b> Multi-Channel Routing .....	47

<b>mETX</b> modified <b>ETX</b> .....	49
<b>MF</b> <b>MF</b> -Transmission <b>Failure</b> .....	40
<b>MIC</b> <b>Metric of Interference and Channel-switching</b> .....	46
<b>MIND</b> <b>Metric of INterference and channel DIVERSITY</b> .....	42
<b>ModWCCETT</b> <b>Modified Weighted Cumulative Consecutive ETT</b> .....	45
<b>MP-OLSR</b> <b>Multiple Path OLSR</b> .....	123
<b>MPR</b> <b>MultiPoint Relays</b> .....	23
<b>MTM</b> <b>Medium Time Metric</b> .....	38
<b>OLSR</b> <b>Optimized Link State Routing Protocol</b> .....	18
<b>OSI</b> <b>Open Systems Interconnection</b>	
<b>PER</b> <b>Packet Error Rate</b> .....	49
<b>PDF</b> <b>Probabiblity Density Function</b> .....	49
<b>QRY</b> <b>QUERY</b> .....	20
<b>RD</b> <b>Random Direction</b> .....	13
<b>RABR</b> <b>Route-Lifetime Assessment Based Routing</b> .....	52
<b>RREQ</b> <b>Route Request</b> .....	19
<b>RREP</b> <b>Route Reply</b> .....	19
<b>RW</b> <b>Random Walk</b> .....	11



<b>RWP Random WayPoint</b> .....	12
<b>SINR Signal-to-Interference-plus-Noise-Ratio</b> .....	42
<b>SNR Signal-to-Noise-Ratio</b> .....	42
<b>SSAR Signal Stability based Adaptive Routing</b> .....	35
<b>TC Topology Control</b> .....	23
<b>TCP Transmission Control Protocol</b> .....	141
<b>TORA Temporary Ordering Routing Algorithm</b> .....	20
<b>UDP User Datagram Protocol</b> .....	141
<b>UMTS Universal Mobile Telecommunication System</b> .....	6
<b>UPD UPDATE</b> .....	20
<b>VANET Vehicular Ad hoc NETWORKs</b> .....	9
<b>WCETT Weighted Cumulative Consecutive ETT</b> .....	46
<b>WCETT Weighted Cumulative ETT</b> .....	44
<b>ZRP Zone Routing Protocol</b> .....	21

# Introduction

## 0.1 Contexte de la thèse

L'évolution récente des technologies de communication sans fil et l'émergence de terminaux mobiles (portables, smartphones, etc.) ont rendu possible l'accès au réseau partout et à tout moment, sans avoir besoin de brancher les appareils communicants à une infrastructure. Un avantage indéniable de ces technologies sans fil est la possibilité "d'être mobile tout en restant connecté". La gestion de la mobilité dans de tels réseaux est alors réalisée de manière efficace et transparente à l'application et à l'utilisateur. La gestion de la mobilité dans un tel réseau est une tâche complexe qui intervient dans différentes couches du modèle OSI.

Dans les réseaux cellulaires ou les réseaux Wi-Fi, les terminaux mobiles sont associés à la meilleure station de base ou au meilleur point d'accès selon la puissance du signal reçue ou d'autres critères de qualité de lien. En cas de mobilité, le terminal voit la puissance de réception de son point d'attachement diminuer et cherche donc un nouveau point d'ancrage avec un signal suffisamment fort. La recherche de ce nouveau point d'ancrage peut être dirigée par le réseau qui examine les mesures remontées par le terminal pour chacune des stations de base du voisinage ou par le terminal lui-même de manière autonome. La mobilité du terminal est appelée "handover" ou transfert intercellulaire et s'effectue au niveau 2 OSI. Si le terminal change en même temps de routeur d'accès, le réseau peut de plus déclencher un re-routage de la connexion, au niveau réseau. Si le terminal était attaché à un réseau Wi-Fi, le changement de point d'accès peut déclencher un changement de sous-réseau IP, des mécanismes de la couche 3 tels que Mobile IP [1] permettent de faire le lien entre ancienne et nouvelle adresse IP.

Ces mécanismes de transfert intercellulaire et de gestion de la mobilité IP ne sont pas directement applicables dans les réseaux ad hoc mobiles. En effet, au lieu de gérer un seul lien, chaque nœud détecte les liaisons sans fil avec tous les nœuds à sa portée radio. En outre, il n'y a pas de changement d'adresse IP à gérer. Le nœud mobile ne change pas d'adresse IP lorsqu'il se déplace, car le réseau ad hoc est un sous-réseau IP. La mobilité des nœuds est alors gérée par le processus de routage plutôt que par les mécanismes de transfert intercellulaire de la couche 2 et de Mobile IP.

Lorsque la mobilité des noeuds est gérée par le protocole de routage, une gestion efficace de cette mobilité peut être réalisée de deux façons : soit par un algorithme de détection de liaison (link sensing) efficace qui peut prévenir des liens défaillants et les invalider avant qu'ils ne soient effectivement inutilisables [2], ou soit par les métriques associées à chaque lien qui peuvent refléter la dégradation de la liaison en raison de la mobilité des noeuds. Une variation de ces métriques implique un changement de route du processus de routage qui remplacera les liens défaillants à des liens plus stables.

Dans les protocoles de routage normalisés par l'IETF, la gestion des liens est faite au niveau 3 et ne se base que sur la bonne ou mauvaise réception de messages de voisinage. Cette limitation est très pénalisante pour la gestion de la mobilité car la complexité et la volatilité de la couche radio font que les mécanismes de niveau 3 tardent à refléter l'état réel des liens. En effet, il y a souvent un décalage entre les états de lien réels et ceux estimés qui est dû à deux problèmes. Le premier problème est provoqué par un décalage temporel de la qualité mesurée qui existe à cause de la fréquence d'envoi/réception de sondes utilisées pour gérer la liaison. Le deuxième problème est dû aux erreurs de mesure qui sont causées par la non-remontée d'informations des couches inférieures. Les métriques obtenues à partir des informations des états de lien souffrent de ces deux problèmes. De plus, la diffusion de ces métriques par le protocole de routage peut causer un délai appréciable entre le moment où les états de liaison sont estimés et leur inclusion dans les calculs de route. Toutes ces incertitudes peuvent conduire à des taux de perte élevés de paquets en présence de mobilité. De nouvelles méthodes de calcul des métriques de routage doivent donc être proposées pour mieux gérer la mobilité. C'est ce que nous nous proposons de faire dans cette thèse.

La première partie de cette thèse porte sur l'étude de l'impact de la mobilité sur les métriques de routage dans les MANET. Dans un premier temps, nous dressons une classification des différentes métriques et établissons celles qui seront prises en compte dans notre approche. Nous considérons les métriques suivantes : Hop Count [3], Expected Transmission Count (ETX) [4], Expected Transmission Time (ETT) [5] et Link Duration (LD) [6]. Hop Count (le nombre de sauts ou de traversée de routeurs) est la plus utilisée par la plupart des protocoles de routage notamment dans les réseaux filaires. Minimiser le nombre de sauts dans une route est justifié par le fait que chaque paquet qui traverse un routeur est traité et mis dans une file d'attente avant d'être retransmis vers le routeur suivant ; un délai important peut donc être introduit.

Une grande partie des métriques de routage sont des dérivées des métriques de base ETX et ETT qui reflètent la qualité de lien. La dernière métrique est LD qui est considérée comme une métrique sensible à la mobilité. Ensuite, une étude comparative entre Hop Count, ETX, ETT et LD est effectuée pour étudier l'impact de la mobilité sur leurs performances. Les travaux de cette thèse se focalisent sur la proposition de nouvelles méthodes de calcul des métriques de routage pour gérer le problème de la mobilité dans les réseaux ad hoc. Nous introduisons de

nouvelles méthodes pour anticiper les valeurs de ces métriques. Notre idée principale consiste à utiliser des algorithmes de prédiction. Ce type de technique est déjà utilisée dans les réseaux avec infrastructure mais, à notre connaissance, la technique d'anticipation de métriques de lien n'a pas été encore proposée dans les réseaux ad hoc. Nous considérons les métriques classiques, en particulier ETX et ETT, mais nous combinons leur calcul à notre algorithme de prédiction.

La simulation est l'approche la plus utilisée comme validation de notre approche. Une évaluation d'une approche par simulation est intéressante mais pas suffisante. Cependant, une évaluation au travers d'un véritable banc d'essai est nécessaire pour valider notre approche. La mise en oeuvre des nouvelles métriques de routage dans un testbed est donc présentée dans la dernière partie de la thèse.

## 0.2 Organisation du mémoire

Les principales contributions de cette thèse sont :

- la proposition de nouvelles métriques de routage qui reflètent la qualité du lien et gèrent la mobilité en même temps dans les réseaux mobiles ad hoc.
- la mise en oeuvre d'un testbed avec ces nouvelles métriques de routage.

Le manuscrit est organisé de la manière suivante :

Le premier chapitre est un chapitre introductif sur les MANET et définit les réseaux mobiles ad hoc, leurs applications, ainsi que les contraintes actuelles telles que la gestion de la mobilité dans ce type de réseau. Ce chapitre introduit aussi un ensemble de protocoles de routage utilisés dans les MANET tel que AODV, DSR, DSDV, OLSR, etc. Notre étude se focalise sur ce dernier protocole (OLSR). Ce choix est justifié dans ce même chapitre.

Le deuxième chapitre dresse un état de l'art sur les métriques de routage existantes dans la littérature et classe ces métriques selon différentes catégories. À partir de cette classification, nous avons déterminé notre choix des métriques de routage à étudier. Le troisième chapitre est consacré à la mise en oeuvre et à l'évaluation de ces métriques afin de comparer leurs performances en présence de mobilité.

Le quatrième chapitre propose deux algorithmes d'anticipation des métriques de routage. Nous y présentons tout d'abord un premier algorithme d'anticipation basé sur un développement de Taylor. Notre second algorithme est une amélioration du premier algorithme car il est basé sur une fonction de prédiction de la future valeur de la puissance du signal reçue. Nous verrons que cet algorithme est plus précis que le premier. Ce chapitre comporte aussi une évaluation des performances des deux algorithmes à partir des simulations montrant l'efficacité des nouvelles métriques par rapport aux métriques classiques.

Le cinquième chapitre est consacré à l'implémentation d'une expérimentation réelle. Ce testbed sert de validation réelle des performances des nouvelles métriques. Ce chapitre décrit l'implémentation des nouvelles métriques, les scénarios d'expérimentations et les résultats.

Le dernier chapitre, la conclusion, dresse un bilan de nos travaux et donne quelques pistes pour des travaux futurs.

### 0.3 Liste des publications

Les contributions de cette thèse ont fait l'objet de trois articles de conférences publiés et un journal soumis.

#### Conférences internationales avec actes

- Sabrine Naimi, Anthony Busson, Véronique Vèque, Larbi Ben Hadj Slama, Ridha Bouallegue : Anticipation of ETX Metric to manage Mobility in Ad Hoc Wireless Networks. The 13th International Conference, ADHOC-NOW 2014, Benidorm, Spain, June 22-27, 2014. Ad-hoc, Mobile, and Wireless Networks Lecture Notes in Computer Science Volume 8487, 2014, pp 29-42.
- Sabrine Naimi, Anthony Busson, Véronique Vèque, Ridha Bouallegue, Larbi Ben Hadj Slama : Mobility management in ad hoc networks using routing metrics. The fourth International Conference on Communications and Networking (ComNet), Mar 2014, Hammamet, Tunisia. pp.1-6, 2014.
- Sabrine Naimi, Anthony Busson, Véronique Vèque, Larbi Ben Hadj Slama, Ridha Bouallegue : Metric Anticipation in Mobile Ad-Hoc Networks. The International Conference on Information Processing and Wireless Systems (IP-WiS), Mar 2013, Djerba, Tunisia.

#### Article de Revue

Le dernier chapitre fait l'objet d'un article journal en cours de préparation pour soumission :

- Sabrina Naimi, Anthony Busson, Véronique Vèque, Ridha Bouallegue : Performance of OLSR with ETX\_ANT in a MANET Testbed. En préparation pour soumission au "Journal of Ad Hoc Networks".

# Chapitre 1

## Les réseaux mobiles ad hoc

### Sommaire

---

<b>1.1</b>	<b>Les réseaux mobiles ad hoc</b>	<b>7</b>
1.1.1	Définition des réseaux mobiles ad hoc	7
1.1.2	Applications des réseaux mobiles ad hoc	8
<b>1.2</b>	<b>Mobilité dans les réseaux ad hoc</b>	<b>9</b>
1.2.1	Définition	9
1.2.2	Modèles de mobilité pour les réseaux ad hoc	11
1.2.2.1	Random Walk	11
1.2.2.2	Random WayPoint	12
1.2.2.3	Random Direction	13
1.2.2.4	Gauss-Markov	13
1.2.2.5	Constant Velocity	14
<b>1.3</b>	<b>Protocoles de routage dans les réseaux ad hoc</b>	<b>15</b>
1.3.1	Problème du routage dans les réseaux ad hoc	15
1.3.2	Classification des protocoles de routage	15
1.3.2.1	Les protocoles de routage proactifs	16
1.3.2.2	Les protocoles de routage réactifs	18
1.3.2.3	Les protocoles de routage hybrides	21
1.3.2.4	Classification	22
<b>1.4</b>	<b>Le protocole de routage OLSR</b>	<b>23</b>
1.4.1	Découverte des voisins	24
1.4.2	Découverte de la topologie	26
1.4.3	Choix du protocole de routage OLSR	27
<b>1.5</b>	<b>Conclusion</b>	<b>28</b>

---

Les technologies sans fil offrent actuellement des perspectives importantes en télécommunications. Les réseaux sans fil présentent un moyen de communication sans fil qui permet la transmission de l'information à travers des unités portables indépendamment de leurs emplacements

et de leur déplacement par le biais d'une interface de communication sans fil. Les environnements mobiles offrent une grande flexibilité d'emploi pour les unités de calcul. Ils permettent une mobilité affranchie aux utilisateurs du réseau sans aucune restriction sur leur localisation. Ces systèmes utilisent des ondes radio plutôt qu'une infrastructure câblée pour communiquer. La mobilité est l'avantage indéniable de ses techniques mais elle engendre des problèmes propres à l'environnement mobile tel que : des ruptures de lien fréquentes (déconnexion), un débit de communication modeste, des ressources limitées en particulier en terme d'énergie.

Les réseaux mobiles sans fil peuvent être classés en deux grandes classes :

- Réseaux sans fil avec infrastructure qui utilisent généralement le modèle de communication cellulaire, tel que **Global System for Mobile communication (GSM)**, **Universal Mobile Telecommunication System (UMTS)** et aujourd'hui **Long Term Evolution (LTE)**. Dans la première classe, une administration centralisée est nécessaire afin d'assurer la communication. Les systèmes de communication cellulaires utilisent essentiellement des réseaux filaires et des stations de bases couvrant les unités mobiles du système. Dans ce mode, deux groupes d'entités distinctes forment le réseau sans fil : les "sites fixes" d'un réseau filaire classique et les "sites mobiles".
- Réseaux sans fil sans infrastructure qui communiquent seulement au travers "des ondes radio", tel que les réseaux ad hoc. Les réseaux mobiles ad hoc sont des réseaux sans infrastructure fixe qui s'auto-organisent rapidement. Les nœuds mobiles forment eux-mêmes une infrastructure du réseau, d'une manière ad hoc. Dans cette classe, le réseau sans fil s'adapte aux différentes conditions de trafic, de propagation et des mouvements dus à la mobilité des nœuds. La taille du réseau peut contenir des centaines ou des milliers de nœuds mobiles : aucune restriction n'est faite sur la taille du réseau. L'évolution des réseaux mobiles ad hoc avec l'utilisation des unités de calculs portables (laptop, smartphone, etc.) offre actuellement "l'accès à l'information n'importe où et n'importe quand", tout en étant mobile [7].

Ce chapitre a pour but de définir les réseaux mobiles ad hoc, et de présenter ses principales caractéristiques et applications. Après la description des différents modèles de mobilité utilisés dans l'environnement ad hoc, nous mettons l'accent sur le problème du routage, problème propre à l'environnement des réseaux mobiles ad hoc. Nous soulignons les principales contraintes que le processus de routage doit considérer et les problèmes associés. Une présentation des différents protocoles de routage existants dans le contexte des réseaux ad hoc est par la suite discutée. Le dernier paragraphe est consacré à la description des principales caractéristiques et fonctionnalités du protocole de routage OLSR : le protocole sur lequel se base notre étude.

## 1.1 Les réseaux mobiles ad hoc

### 1.1.1 Définition des réseaux mobiles ad hoc

Un réseau mobile ad hoc, appelé généralement **Mobile Ad hoc NETWORK (MANET)**, est un ensemble de nœuds mobiles qui se déplacent dans un territoire quelconque d'une manière autonome et coopérative, sans l'utilisation d'une infrastructure préexistante ou d'une administration centralisée. Les "ondes radio" qui se propagent entre les différents nœuds mobiles sont le seul moyen de communication. Dès qu'un ensemble de nœuds mobiles se trouve à portée radio les uns des autres, alors le réseau se forme spontanément mais de manière provisoire. Il existe deux modes de communication entre deux nœuds mobiles qui dépendent de la distance qui les sépare. Dans le cas où les deux nœuds sont à portée radio et peuvent communiquer directement, ce mode est appelé transmission ad hoc. En revanche, dans le cas où les deux nœuds ne sont pas à portée, ils doivent utiliser d'autres nœuds mobiles comme relais afin d'assurer la communication et d'acheminer les paquets à destination, ce mode est appelé transmission multi-sauts [7].

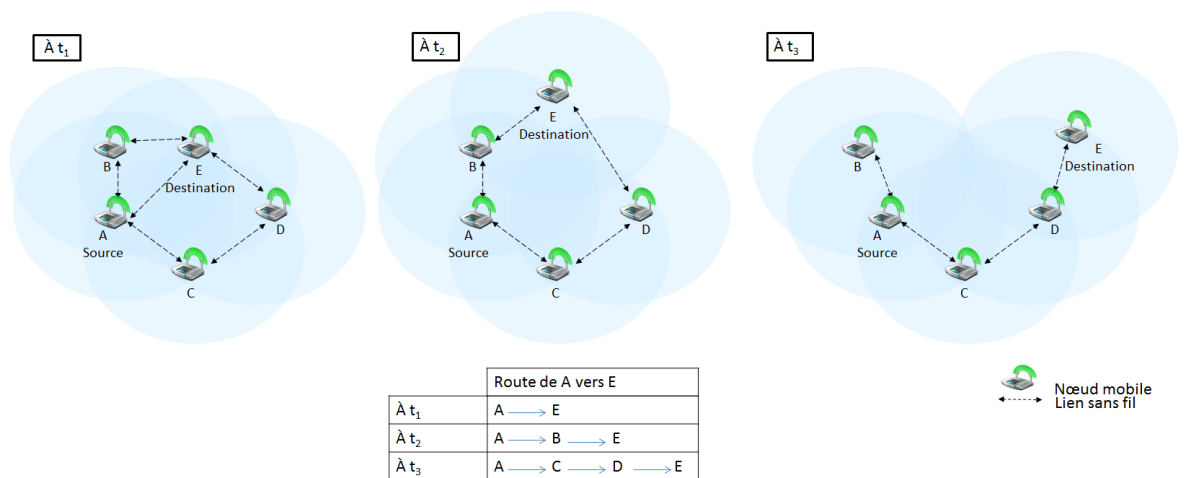


FIGURE 1.1 – Changement de la topologie à cause de la mobilité

La figure 1.1 montre un exemple de changement de la topologie à cause de la mobilité des nœuds sans fil. Chaque fois qu'un nœud bouge, les tables de routage doivent être recalculées pour prendre en compte les mises à jour. À  $t_1$ , nous observons dans la figure 1.1 que la route de A vers E est le lien direct  $A \rightarrow E$ , si nous considérons le nombre de sauts comme métrique de routage. À  $t_2$ , la topologie change : E n'est plus à portée radio avec A, alors le lien n'existe plus. De ce fait, la route de A vers E change et devient :  $A \rightarrow B \rightarrow E$ . Les nœuds bougent encore à  $t_3$ , E s'éloigne de B alors la route qui existe pour l'atteindre devient  $A \rightarrow C \rightarrow D \rightarrow E$ .

La mobilité des terminaux est l'avantage indéniable des réseaux mobiles ad hoc. La topologie du réseau peut changer à tout moment d'une manière dynamique, rapide et aléatoire. Mais, les



changements fréquents de la topologie peuvent engendrer la rupture des liens. En plus, l'utilisation des "ondes radio" pour communiquer limite la bande passante réservée à un nœud. Les erreurs de transmission radio sont plus fréquentes dans les réseaux mobiles ad hoc que dans les réseaux filaires. Une autre contrainte s'ajoutant aux contraintes des réseaux mobiles ad hoc est celle de l'interférence. Deux transmissions simultanées sur une même fréquence ou utilisant des fréquences proches peuvent interférer. Aussi, les contraintes et limitations physiques qui minimisent le contrôle des données transférées sont les causes d'une fiabilité limitée dans ce type de réseau.

### 1.1.2 Applications des réseaux mobiles ad hoc

D'une façon générale, les MANET sont utilisés dans toutes applications où le déploiement d'une infrastructure réseau filaire est trop contraignant. Les applications dédiées aux MANET couvrent un très large spectre. La figure 1.2 présente quelques exemples d'applications MANET. Ils sont caractérisés par l'absence d'une infrastructure préexistante. Une liste des applications est donnée ci-dessous [8].

- Les environnements militaires ou les champs de bataille demeurent les toutes premières applications des réseaux ad hoc. Les nœuds mobiles de ce type de réseau peuvent être des soldats, des chars, des avions de chasse, etc. Pour les applications militaires, les MANET peuvent être utilisés afin de surveiller le déplacement de l'ennemi dans un espace géographique donné, ou de récolter des données dans une zone dangereuse où l'envoi d'êtres humains serait risqué.
- L'usage personnel utilisé pour le partage des applications des équipements mobiles et des objets domestiques tels que smartphones, laptops, imprimantes, etc. Mais, les MANET sont déployés aussi pour contribuer au confort domestique en transformant les logements personnels en environnements intelligents qui s'adaptent automatiquement au comportement des utilisateurs et évitent le câblage à la maison.
- Les interventions d'urgence, opérations de recherche, de secourisme et de sauvetage en urgence durant les catastrophes naturelles comme lors du tremblement de terre, feux, inondation, etc.
- Les environnements civils déployés pour permettre le travail collaboratif et les communications au sein d'une salle de conférence, d'un campus universitaires, des stades, des bateaux, etc.
- Applications commerciales utilisés pour un paiement électronique distant (taxi, boutiques) ou pour un accès Internet mobile, etc.
- Réseaux de capteurs généralement exploités pour des applications environnementales (météo, activité terrestre, suivi animale, etc.). Leur usage permet l'analyse et la gestion

de phénomènes complexes sur une longue période de temps et sur une large zone géographique tel que : la température, l'humidité, la pression, le bruit, etc.

- Réseaux mobiles fortement un réseau de véhicules communicants (**V**ehicular **A**d hoc **N**ETworks (**VANET**)) présente un exemple de ce type. **VANET** est un groupe de véhicules à portée les uns des autres formant des systèmes de transports intelligents. L'intérêt optimal des VANET est de contribuer à des routes plus sûres et plus efficaces dans le futur en fournissant des informations opportunes aux conducteurs et aux autorités intéressées.
- Réseaux Mesh : permet d'étendre la portée d'un réseau (la couverture radio à plusieurs usagers) ou de le densifier : environnement civil déployé.

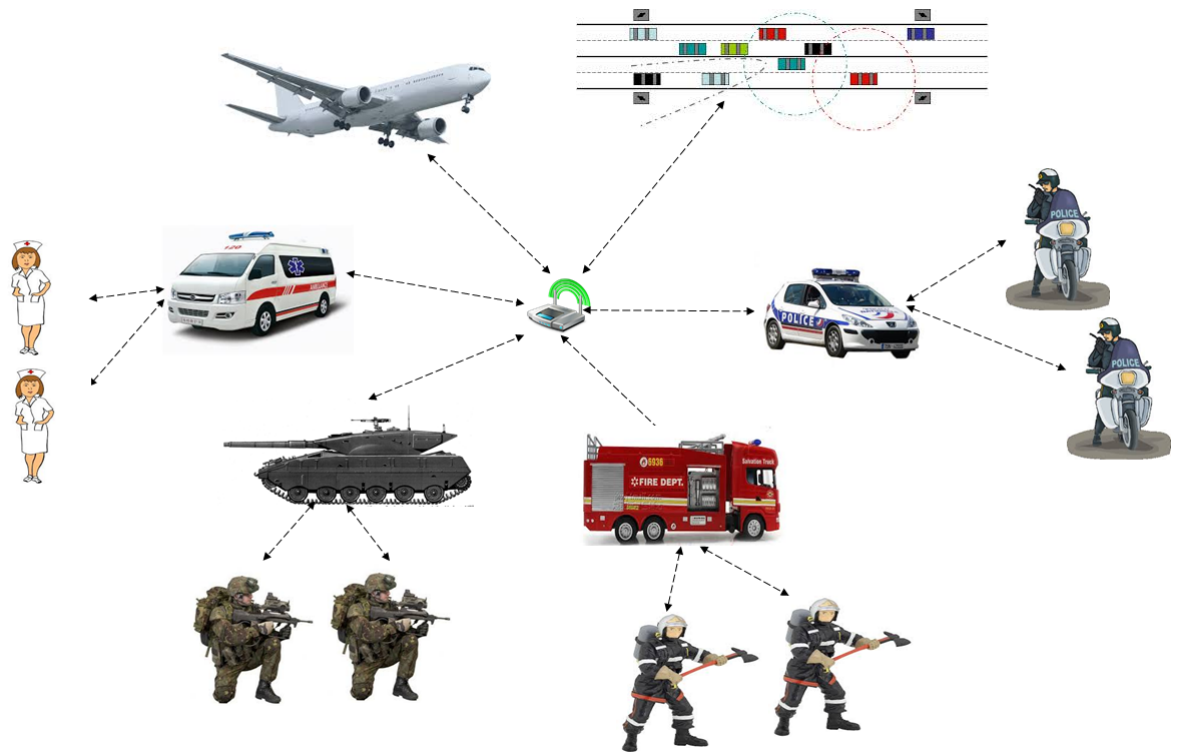


FIGURE 1.2 – Exemples d'applications MANET

## 1.2 Mobilité dans les réseaux ad hoc

### 1.2.1 Définition

Le principal avantage des MANET est la possibilité de maintenir une communication tout en étant mobile. Mais, cet avantage de mobilité dans les MANET présente aussi une contrainte pour la gestion du réseau. Des changements fréquents et dynamiques dans la topologie du réseau sont provoqués par la mobilité des nœuds. Des routes peuvent se créer et disparaître très souvent, et d'une manière brusque, durant des intervalles de temps variables. Ces déplacements

ont naturellement un impact sur le réseau. Ils peuvent modifier la topologie du réseau ainsi que le comportement du canal de communication sans fil. Pour une bonne gestion du réseau, ces modifications doivent être prises en compte le plus rapidement possible par le protocole de routage afin d'assurer une vision correcte de la topologie des nœuds. La réactivité du protocole de routage face à ces modifications a un impact direct sur la qualité des routes. Le protocole de routage utilise les métriques de routage pour sélectionner la meilleure route. Dans le cas d'une mobilité brusque ou rapide, il y aura un délai entre les valeurs des métriques réelles et celles prises en compte par le protocole de routage. Cela implique des pertes de paquets car les tables de routage ne sont pas mises à jour suffisamment à temps, avec les bonnes valeurs de métriques.

Prenons l'exemple de la figure 1.3 pour mieux illustrer le délai généré entre la rupture d'un lien et la prise en compte par le protocole de routage. Nous observons qu'à un temps  $t_1$ , la route de A vers E est  $A \rightarrow E$ , suivant un lien direct, c'est cette route qui va être prise en compte par le protocole de routage. Considérons, qu'à  $t_1 + \Delta t$ , le nœud E bouge et s'éloigne brusquement de telle sorte qu'il n'existe plus de lien direct entre A et E. Mais, le protocole de routage considère toujours la route  $A \rightarrow E$ , à cause du délai qui existe entre la rupture d'un lien et la prise en compte par le protocole de routage. De cette manière les paquets sont perdus.

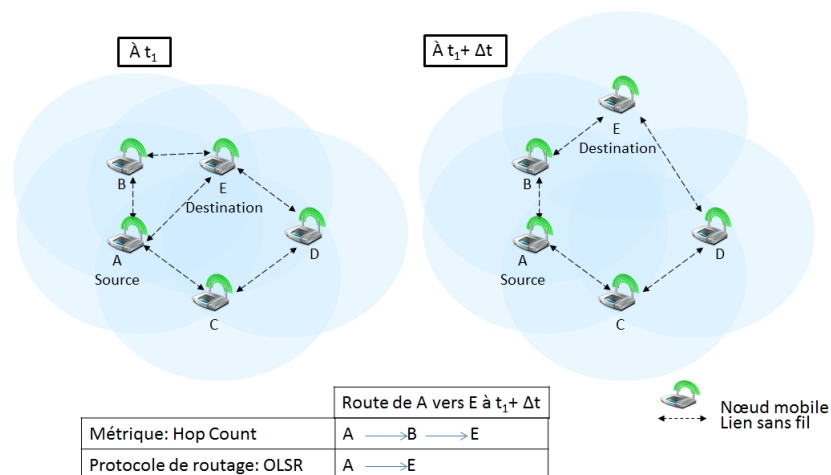


FIGURE 1.3 – Délai généré à cause de la mobilité dans la route considérée par le protocole de routage

La mobilité est un phénomène intrinsèquement aléatoire qui influe sur la qualité des communications sans fil. La modélisation des communications sans fil est très complexe, d'une manière générale. En plus de la mobilité, il est nécessaire de modéliser le système physique, d'une part et de modéliser aussi les interactions entre les protocoles des différentes couches lors de la communication, d'autre part.

Par exemple dans un environnement urbain, la mobilité des terminaux dépend uniquement de la mobilité des personnes qui les transportent. La mobilité impacte les performances des algorithmes chargés d'assurer les échanges de données au sein du réseau : en particulier le protocole

de routage. Pour cette raison, il est primordial de disposer de modèles de mobilité selon le contexte de l'étude.

### 1.2.2 Modèles de mobilité pour les réseaux ad hoc

Les performances des MANET sont souvent analysées par des simulations, ainsi l'évaluation des protocoles de routage ou de nouveaux algorithmes dépend des paramètres de simulation pris en considération. Il est nécessaire que les simulations reproduisent soigneusement les conditions du fonctionnement réel du protocole de routage pour que les résultats de cette évaluation de performance soient pertinents. De nombreux paramètres sont ainsi pris en compte par l'environnement de simulation, surtout en cas de mobilité tels que : la vitesse des nœuds mobiles, la direction de leurs mouvements, le temps de pause, etc. Le simulateur doit donc considérer ces paramètres à l'aide d'un modèle de mobilité décrivant le type de déplacement dans le réseau [9].

Au cours de ces dernières années, de nombreux modèles de mobilité ont été proposés dans la littérature afin de simuler le comportement des nœuds. Dans le paragraphe suivant, nous allons donc étudier les différents modèles de mobilité utilisés dans les environnements de simulation.

#### 1.2.2.1 Random Walk

Le modèle de mobilité **Random Walk (RW)** a été décrit mathématiquement, pour la première fois par Einstein en 1926. Dans ce modèle, chaque nœud mobile choisit aléatoirement un angle de direction et une vitesse de mobilité. La valeur de l'angle de direction varie dans l'intervalle  $]0, 2\pi]$ , et la valeur de la vitesse de mobilité varie sur un intervalle  $[V_{min}, V_{max}]$ . Les deux paramètres  $V_{min}$  et  $V_{max}$  sont deux constantes représentant respectivement la vitesse minimale et la vitesse maximale de déplacement d'un mobile.

Le déplacement du mobile se fait pendant une période de temps  $t$  ou une distance  $d$ . Une fois arrivé à sa destination, le nœud se déplace à nouveau suivant une nouvelle direction et une nouvelle vitesse. Ces changements de direction et de vitesse lors du nouveau déplacement sont absolument aléatoires et indépendants du choix précédent.

Bien qu'à l'origine, le modèle Random Walk n'était pas dédié pour représenter le déplacement de mobiles dans un réseau ad hoc, de nombreuses études l'utilisent dans l'évaluation de performance des protocoles de communication [11]. Un exemple de trajectoire d'un mobile généré par le modèle de mobilité Random Walk est illustré dans la figure 1.4.

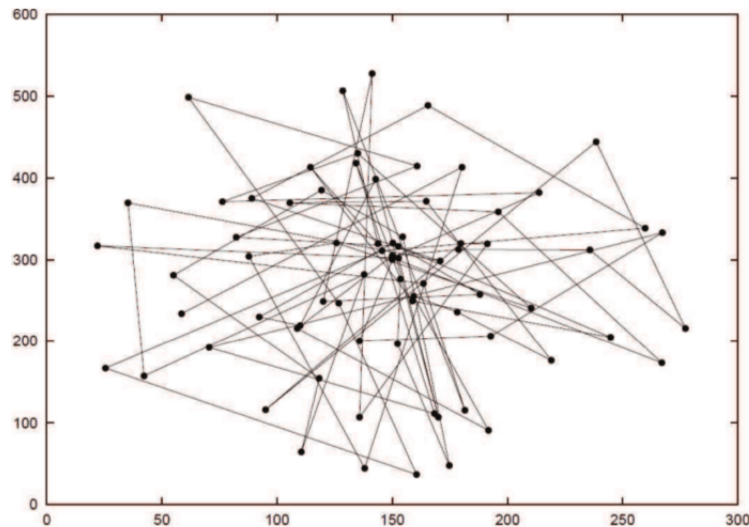


FIGURE 1.4 – Trajectoire d'un mobile générée par le modèle Random Walk [10]

### 1.2.2.2 Random WayPoint

Le modèle Random Waypoint **Random WayPoint (RWP)** a été introduit pour la première fois dans [12], puis il a été raffiné. La mobilité des nœuds est typiquement aléatoire et tous les mobiles sont distribués uniformément dans l'espace de simulation. Afin de modéliser la trajectoire d'un mobile, RWP prévoit trois paramètres : sa destination, sa vitesse et son temps de pause. Le changement de vecteur vitesse est événementiel : il survient à chaque fois qu'un mobile atteint sa destination, avec un temps de pause entre chaque déplacement. L'algorithme RWP consiste à :

- placer un certain nombre de mobiles dans un espace de simulation (une zone carrée),
- affecter une position de coordonnées  $(x,y)$ , une vitesse de déplacement, et une destination initiale à atteindre pour chaque mobile,
- faire repartir vers chaque mobile une autre destination choisie aléatoirement avec une nouvelle vitesse après un éventuel temps de pause et à chaque fois que les mobiles atteignent leur destination dans le carré,
- répéter le même processus.

Il existe des points de ressemblances et de différences entre les deux modèles de mobilité RW et RWP. La principale différence entre ces deux modèles est la fréquence des changements des vecteurs vitesse qui est périodique avec RW et événementielle avec RWP. Par contre, la norme du vecteur vitesse est identique pour les deux modèles, et choisie uniformément sur  $[V_{min}, V_{max}]$ . La direction est choisie sur l'intervalle  $]0, 2\pi]$  pour les deux modèles, mais dans RW, elle est définie uniformément, en revanche dans RWP, elle est déterminée par sa destination. Un exemple de trajectoire d'un mobile généré par le modèle de mobilité Random Walk est présenté dans la figure 1.5. Du fait de la simplicité de ce modèle, RWP est le modèle le plus utilisé dans les

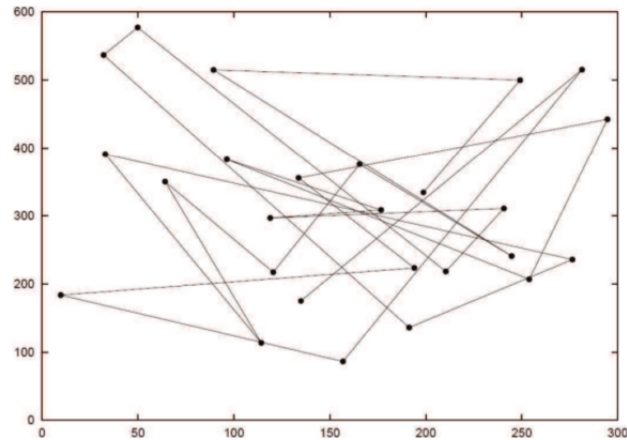


FIGURE 1.5 – Trajectoire d'un mobile générée par le modèle Random WayPoint [10]

travaux d'évaluation dans les MANET [13]. Cependant, ce modèle n'est pas toujours adapté pour décrire une mobilité complexe.

### 1.2.2.3 Random Direction

Pour améliorer le modèle RWP, les auteurs de [14] ont proposé le modèle de mobilité **Random Direction (RD)** dont le fonctionnement est le suivant :

- placement de l'ensemble des mobiles dans la zone de simulation en choisissant leur position uniformément répartie suivant les deux directions de l'espace,
- Chaque mobile choisit uniformément une direction dans  $[0, 2\pi]$ , et une vitesse dans  $[V_{min}, V_{max}]$ ,
- déplacement des mobiles selon la direction et la vitesse choisies jusqu'à atteindre le bord de la zone de simulation,
- une fois qu'un mobile ayant atteint cette bordure, il choisit une nouvelle direction avec une nouvelle vitesse après avoir effectué une pause dont la durée de temps est choisie uniformément dans  $[P_{min}, P_{max}]$ ,

Le modèle RD ne concentre pas les mobiles au milieu de la zone de simulation par rapport au modèle RWP comme le montre l'exemple de trajectoire généré par le modèle de mobilité RD illustré dans la figure 1.6.

### 1.2.2.4 Gauss-Markov

Le modèle de mobilité Gauss-Markov [15] est purement aléatoire. Les changements de direction se font d'une manière instantanée et les changements de vitesse passent brusquement d'une valeur à une autre. Dans ce modèle, le temps de la simulation est discrétisé à des durées

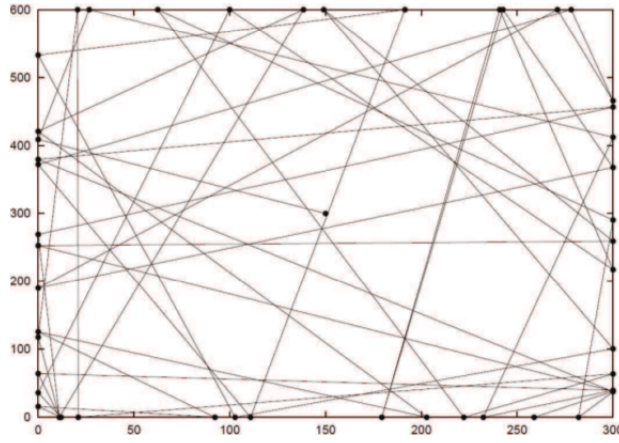


FIGURE 1.6 – Trajectoire d'un mobile générée par le modèle Random Direction [10]

constantes. Un nouveau vecteur vitesse est choisi et noté  $\vec{v}_t$ , pour chaque instant  $t$ . Le processus de ce modèle est Markovien car son état futur dépend uniquement de son état présent (il ne dépend pas des états passés). Afin de pouvoir introduire une modification entre deux valeurs consécutives de vitesse, ce processus fait intervenir une distribution normale. Il contient donc une composante gaussienne. Dans un espace à deux dimensions, le vecteur vitesse est exprimé par :

$$\vec{v}_{t+1} = \begin{cases} v_x^{t+1} = \alpha v_x^t + (1 - \alpha)\mu_x + \sigma_x \sqrt{1 - \alpha^2} w_x^t \\ v_y^{t+1} = \alpha v_y^t + (1 - \alpha)\mu_y + \sigma_y \sqrt{1 - \alpha^2} w_y^t \end{cases} \quad (1.2.1)$$

Où  $\vec{\mu} = [\mu_x, \mu_y]$  est le vecteur qui représente la vitesse moyenne et  $\vec{\sigma} = [\sigma_x, \sigma_y]$  est son écart type.  $\vec{w} = [w_x, w_y]$  exprime le vecteur d'un processus gaussien normalisé  $N(0, 1)$ . Le vecteur vitesse  $\vec{v}_{t+1}$  à un instant  $t + 1$ , dépend de deux composantes à la fois : la première composante est sa vitesse  $\vec{v}_t$  à l'instant  $t$ , et la deuxième composante totalement indépendante est  $\vec{w}_t$ .

Afin d'établir une dépendance temporelle au niveau du vecteur vitesse, la constante  $\alpha \in [0, 1]$  est utilisée. Ce facteur influence le vecteur vitesse. Plus  $\alpha$  est proche de 1, plus  $\vec{v}_{t+1}$  est proche de  $\vec{v}_t$ . Dans le cas où  $\alpha = 1$ , le vecteur vitesse dans le modèle Gauss-Markov devient constant. En revanche, plus  $\alpha$  est proche de 0, et plus le vecteur vitesse subit des oscillations importantes. Dans le cas où  $\alpha = 0$ ,  $\vec{v}_{t+1}$  et  $\vec{v}_t$  sont totalement indépendants. Alors, la trajectoire des mobiles générée par le modèle Gauss-Markov est la même trajectoire générée par le modèle Random-Walk décrit dans 1.2.2.1.

### 1.2.2.5 Constant Velocity

Dans le modèle simple Constant Velocity (CV) [16], les nœuds sont placés d'une manière aléatoire sur un plan infini, avec une densité finie. Les nœuds se déplacent linéairement à une vitesse constante dans des directions aléatoires, mais ils ne changent pas leurs directions en

cours de déplacement.

Après avoir présenté les modèles de mobilités les plus utilisés pour l'évaluation des protocoles de routage, nous allons étudier dans la section suivante, les protocoles de routages des MANET et leur classification.

## 1.3 Protocoles de routage dans les réseaux ad hoc

### 1.3.1 Problème du routage dans les réseaux ad hoc

Le routage doit assurer une stratégie d'acheminement de données, à n'importe quel moment, entre une paire de nœuds (source et destination) à travers un réseau. La problématique de routage présente l'une des principales problématiques soulevées par les communications radio multi-sauts. Dans les MANET, les mobiles sont obligés d'assurer le routage des données par eux-mêmes, à cause de l'absence d'infrastructure dans ce type de réseau. Lorsque la destination est hors portée radio de la source, chaque mobile du réseau sert de relais en retransmettant les données vers un autre mobile jusqu'à atteindre la destination. L'instabilité fréquente des chemins à cause de la mobilité et les contraintes de ressources radio (débit variable, bande passante limitée, etc.) présentent des défis majeurs pour ce type de routage.

Les protocoles de routage assurent le transfert des données d'un mobile à un autre à travers des routes multi-sauts. Les liens radio changent d'une manière dynamique et fréquente : les liens se créent et se détruisent à cause du déplacement libre des mobiles. Les protocoles de routage s'adaptent à ces changements de routes. En plus, ils doivent prendre en considération les contraintes des ressources limitées qui caractérisent les MANET. Le protocole de routage doit assurer le routage des données à travers des routes disposant des ressources nécessaires malgré le changement fréquent des chemins. Le groupe de travail MANET de l'Internet Engineering Task Force ([IETF](#)) a normalisé un certain nombre de protocoles de routage qui seront décrits par la suite.

### 1.3.2 Classification des protocoles de routage

Le principal but de toute stratégie de routage est de mettre en œuvre une méthode de gestion d'acheminement robuste et efficace. En général, toute stratégie de routage repose sur des mécanismes que nous pouvons regrouper en trois grandes classes : les protocoles de routage proactifs, les protocoles de routage réactifs et les protocoles de routage hybrides. Mais, il existe d'autres types de classification tels que : géographiques, hiérarchiques, à qualité de service et multicast [[17](#)], etc.



Dans notre étude, nous nous sommes intéressés seulement aux protocoles de routage unicast sans garantie de qualité de service.

### 1.3.2.1 Les protocoles de routage proactifs

Les protocoles de routage proactifs diffusent des messages de contrôle périodiques pour la création, la maintenance et la mise à jour des routes, et ce en parallèle de la transmission des données. Même s'il n'y a pas de trafic, cette maintenance reste toujours active. Cette classe de protocole est basée sur les mêmes méthodes utilisées pour les protocoles de routage dans les réseaux filaires. Les deux principales méthodes sont : la méthode à État de Liens (Link State) et la méthode du Vecteur de Distances (Distance Vector). Ces deux méthodes reposent sur la technique de plus court chemin ce qui permet à une source de trouver le chemin le plus court vers une destination.

- **La méthode à État de Liens (Link State)** Dans cette méthode, chaque nœud maintient sa propre vision de toute la topologie du réseau. La mise à jour de cette vision se fait par diffusion périodique (par inondation) des requêtes par chaque nœud déclarant l'état des liens de ses voisins à tous les nœuds du réseau. L'opération de mise à jour peut se faire aussi dans le cas de changement d'un état des liens. Une fois que la mise à jour est effectuée, chaque nœud change sa vision de la topologie en se basant sur l'image complète du réseau formé des liens les plus récents. Ensuite, il applique un algorithme de calcul de route optimale pour calculer la distance qui le sépare d'une destination donnée. L'algorithme Dijkstra [18] est un exemple des algorithmes les plus couramment appliqués dans le calcul de plus court chemin.
- **La méthode du Vecteur de Distances (Distance Vector)** Par contre dans cette méthode, chaque nœud diffuse à ses voisins sa vision des distances qui le séparent de tous les hôtes du réseau. Chaque nœud calcule le chemin le plus court vers n'importe quelle destination en se basant sur les informations reçues par tous ses voisins. Cette approche est basée sur l'algorithme distribué de Bellman Ford (BF) [19]. Le processus de calcul de plus court chemin se répète jusqu'à ce que le réseau atteigne un état stable.

Les protocoles de routage proactifs réunissent les idées des deux méthodes. Ils essayent de réduire leurs limitations, et de prendre en compte les caractéristiques des environnements mobiles afin de les adapter à ce nouvel environnement.

### Exemples de protocoles de routage proactifs

Dans cette catégorie, les protocoles de routage proactifs maintiennent une vision de la table de routage dans chaque nœud, grâce aux messages de mise à jour communiqués aux nœuds à

chaque changement de la topologie. Les protocoles basés sur ce principe, dits aussi à diffusion de table, sont entre autres : DSDV [20], WRP [21], GSR [22], FSR [23], HSR [24], ZHLS [25], CGSR [26], DREAM [27], OLSR [28], etc.

Dans la suite de ce paragraphe, nous détaillons :

- Le protocole **D**ynamic **d**estination **S**equenced **D**istance **V**ector (**DSDV**) [20] a été conçu spécialement pour les réseaux mobiles. Cet algorithme est dérivé de l'algorithme de vecteur de distances basé sur **BF** [19]. Mais, DSDV a été proposé dans le but d'éviter le problème de boucles qui peuvent apparaître avec les algorithmes de vecteur de distances. Pour commander l'information de routage, DSDV étiquette chaque entrée de la table de routage avec un numéro de séquence, permettant de distinguer les nouvelles routes des anciennes. Un des avantages principaux de DSDV est qu'il fournit des routes sans boucles, cependant, DSDV est très lent [29].
- Le protocole **G**lobal **S**tate **R**outing (**GSR**) [22] est similaire à DSDV. Il est basé aussi sur la méthode d'état de liens et il utilise une vue globale du réseau. Mais GSR améliore ces deux algorithmes (l'algorithme DSDV et l'algorithme à état de liens) en évitant l'inondation des messages car il utilise une méthode de dissémination de BF, qui implique l'absence d'inondation. La façon dont les informations de routage circulent dans le réseau est la différence clé entre l'algorithme GSR et l'algorithme à état de liens. La maintenance dans GSR de la table la plus récente est effectuée à partir de l'échange périodique d'état des liens à travers les voisins locaux uniquement [29].
- Le protocole **F**isheye **S**tate **R**outing (**FSR**) [23] est une amélioration du protocole GSR. Il est basé sur la technique "œil de poisson" (fisheye)<sup>1</sup>. Dans GSR, une grande quantité de la bande passante est gaspillée à cause de la grande taille des messages de mise à jour. Par contre dans FSR, les messages de mise à jour contiennent des informations sur les nœuds les plus proches, réduisant ainsi la taille de ces messages. Ainsi, la précision de l'information diminue quand la distance du nœud augmente [29].
- Le protocole **C**lusterhead **G**ateway **S**witch **R**outing (**CGSR**) [26] utilise le protocole DSDV. L'architecture du réseau est basée sur la notion de groupes. Des nœuds spécifiques dans CGSR sont élus pour être des clusterheads qui commandent l'accès au canal dans un groupe au niveau de la couche **M**edium **A**ccess **C**ontrol (**MAC**). Les clusterheads sont destinés prioritaires à transmettre de l'information entre les groupes. Les nœuds d'un groupe communiquent entre eux et avec leur clusterhead. Le protocole CGSR a modifié

---

1. La technique fisheye a été proposée par Kleinrock et Stevens dans le but de réduire le volume d'informations nécessaires dans la présentation des données graphique. Dans le contexte de routage, cette technique capture les données concernant la précision de la distance et la qualité du chemin d'un voisin direct. Cette précision diminue progressivement, quand la distance augmente.

DSDV de sorte qu'il profite de l'architecture du groupe pour effectuer le routage hiérarchique. Ce type de routage offre l'avantage de limiter la maintenance des routes aux clusterhead. Mais, la maintenance des groupes induit une charge supplémentaire [29].

- Le protocole **Hierarchical State Routing (HSR)** [24] utilise aussi les notions de partitionnement et de groupes. Le réseau est divisé en groupes, et chaque groupe possède un représentant élu. Même les représentants des groupes s'organisent en groupes et ainsi de suite. Dans chaque groupe physique, les nœuds diffusent leurs informations de liens entre eux. Le représentant du groupe récapitule l'information de son groupe et l'envoie aux représentants des groupes voisins. La topologie est vue comme une hiérarchie de plusieurs niveaux : les représentants des groupes sont membres du groupe sur un plus haut niveau, les nœuds d'un groupe diffusent des informations plus élémentaires à un niveau plus bas et ainsi de suite. La perte des paquets envoyés lorsqu'une route devient invalide présente l'un des principaux inconvénients de HSR. Aussi, HSR utilise des adresses hiérarchiques longues. Cependant, le changement fréquent des adresses rend difficile la tâche de la localisation des nœuds [29].
- Le protocole **Distance Routing Effect Algorithm for Mobility (DREAM)** [27] destiné aux réseaux ad hoc mobiles utilise les informations de localisation des unités mobiles dans le routage comme les coordonnées géographiques. DREAM effectue une inondation (propagation partielle) pour diffuser les données spécifiques à une destination donnée. Chaque entité du réseau échange des messages de contrôle périodiquement pour informer les autres entités de sa localisation. Ces messages de contrôle sont envoyés aux nœuds les plus proches. La distance qui sépare les nœuds influe cet échange. De plus, DREAM essaye de s'adapter à la mobilité des entités du réseau par la fréquence de contrôle de mise à jour qui se base sur les vitesses des mouvements.
- Le protocole **Optimized Link State Routing Protocol (OLSR)** [28] est une optimisation de l'algorithme à état de liens. Ce protocole utilise aussi l'algorithme du plus court chemin pour la sélection des routes. Dans l'algorithme à état de liens, chaque nœud du réseau déclare son voisinage à tout le réseau. Par contre dans OLSR, les nœuds ne déclarent qu'un sous-ensemble de leur voisinage grâce à la technique des relais multipoints. Dans la section 1.4, nous décrivons le fonctionnement d'OLSR, et nous expliquons le choix de ce protocole pour notre étude.

### 1.3.2.2 Les protocoles de routage réactifs

Comme nous l'avons mentionné au cours du paragraphe précédent, les protocoles de routage proactifs essaient de maintenir les meilleurs chemins vers toutes les destinations du réseau par l'échange périodique de messages de contrôle de mise à jour. Les routes sont sauvegardées

même si elles ne sont pas utilisées ce qui induit un contrôle excessif surtout dans le cas des réseaux denses.

Les protocoles de routage réactifs créent et maintiennent les routes selon les besoins des émetteurs et de leurs applications. Dans ce cas, une procédure de découverte globale de routes est lancée qui permet d'avoir une information bien spécifique mais inconnue au préalable. Les protocoles basés sur ce principe, dits aussi à la demande, sont entre autres : DSR [12], AODV [30], CBRP [31], LMR [32], TORA [33], ABR [34], SSR [35], LAR [36], etc.

### Exemples de protocoles de routage réactifs

Dans ce paragraphe, nous présentons quelques exemples de protocoles de routage réactifs :

- Le protocole **Dynamic Source Routing (DSR)** [12] considère le routage à la source, du fait que la source désigne le chemin complet vers la destination dans l'en-tête du paquet. Le long de ce chemin, chaque nœud expédie tout simplement le paquet destiné au prochain saut indiqué dans le chemin grâce à un cache route. Ainsi, une source cherche tout d'abord dans son cache de route pour déterminer la route à la destination. Si elle existe, la source utilise cette route. Sinon, la source lance une procédure de découverte de route pour construire une route. Dans cette procédure de découverte, le nœud source diffuse une requête par inondation dans le réseau. La réponse est retournée par la destination ou un autre nœud si celui-ci a une route dans son cache pour cette destination.

Dans DSR, peu de messages de contrôle circulent dans le réseau quand peu de sources veulent communiquer, ce qui représente l'avantage principal de ce protocole. Mais, dans le cas des réseaux de grande taille, DSR peut présenter des problèmes de passage à l'échelle (scalability). Dans ce cas, les paquets augmentent de taille car ils doivent véhiculer les adresses des nœuds existants dans le chemin et l'inondation pour la recherche de routes devient plus importante.

- Le protocole **Ad Hoc On-Demand Distance Vector (AODV)** [30] est considéré comme une combinaison du protocole DSR et du protocole DSDV. Il combine les mécanismes de découverte et de maintenance de routes de DSR, et les mécanismes d'association de numéro de séquence et de mise à jour périodiques de DSDV.

Lors de la découverte de route, AODV diffuse un message **Route Request (RREQ)**. Ce protocole utilise le numéro de séquence pour éviter le problème de boucle et utilise les routes les plus récentes. Lorsqu'un nœud de transit envoie un RREQ à un voisin, il enregistre la source dans la table de routage à partir de la première copie de la requête reçue. Cette mise à jour sert à la construction du chemin inverse. Une fois que la destination reçoit le message, elle retransmet un **Route Reply (RREP)** vers la source par le chemin inverse.

La mise à jour des routes dans AODV est assurée par l'échange périodique de messages HELLO entre les voisins. Les deux messages RREQ et RREP sont envoyés sur le même chemin, alors AODV ne supporte que les liens symétriques. En plus, AODV n'assure pas le choix du meilleur chemin entre la source et la destination mais il évite le problème de boucle de routage de BF, ce qui offre une convergence rapide quand la topologie du réseau change.

- Le protocole **Cluster Based Routing Protocol (CBRP)** [31] est basé sur la notion de groupes : les nœuds du réseau sont divisés en groupes. Un nœud devient membre d'un groupe par l'échange de messages HELLO. Les chefs de groupe (Clusterheads) sont changés aussi rarement que possibles. Chaque nœud du réseau met à jour une table de voisinage qui contient l'état de chaque voisin (chef du groupe ou membre) et le mode du lien (unidirectionnel ou bidirectionnel). Un chef du groupe sauvegarde des informations sur les membres de son groupe et aussi les groupes voisins en mettant à jour aussi une table d'adjacence du groupe. Chaque entrée dans cette table indique la passerelle qui permet l'accès à chaque groupe et le chef de chaque groupe.

Lors de l'envoi de données, une source diffuse par inondation des paquets de demande de routes par une source, mais seulement aux chefs de groupes voisins. Chaque chef de groupe vérifie si la destination existe dans son groupe. Si c'est le cas, il renvoie la demande directement à la destination, sinon, il l'envoie à tous ses chefs de groupes. Une fois que la destination reçoit le paquet de demande, elle répond avec la route enregistrée dans ce paquet de demande. Alors, CBRP utilise un routage à la source (source routing).

- Le protocole **Temporary Ordering Routing Algorithm (TORA)** [33] utilise une technique de routage appelée "Inversement de Liens" (Link Reversal). Ce protocole est basé sur les graphes acycliques orientés (**Directed Acyclic Graph (DAG)**) pour la création d'une route vers une destination. La source diffuse un message **QUERY (QRY)** spécifiant l'identificateur de la destination, et le récepteur répond par l'envoi d'un message **UPDATE (UPD)**. TORA a été conçu principalement dans le but de minimiser les changements fréquents de la topologie (mobilité) dans les réseaux ad hoc. Ce protocole sauvegarde plusieurs chemins vers une même destination, ce qui fait que les changements fréquents de la topologie n'influencent pas le routage de données, à moins que tous les chemins qui mènent vers la destination soient rompus. Donc, le nombre de messages de contrôle utilisé est réduit dans le réseau.
- Le protocole **Associativity Based Routing (ABR)** [34] est basé sur une nouvelle approche de routage qui utilise une nouvelle métrique connue sous le nom de degré de stabilité d'association. Dans ce protocole, la sélection des routes repose sur les états d'associativité des nœuds. Les routes sélectionnées auront ainsi une longue durée de vie. Les nœuds du réseau signalent leur existence par l'échange de messages de contrôle périodique appelé beacons.

À la réception d'un beacon, le nœud met à jour sa table d'associativité : il incrémente sa valeur d'associativité correspondant au nœud émetteur.

La stabilité d'association est définie comme la stabilité de connexion entre deux nœuds. Une grande valeur d'associativité avec un nœud signifie une faible mobilité. Par contre, une petite valeur d'associativité indique une grande mobilité. Ainsi, la recherche des chemins avec une longue durée de vie est l'objectif principal d'ABR.

Lors de la phase de découverte de route vers une destination, une source diffuse un message **Broadcast Query (BQ)**. Un nœud réachemine le BQ reçu en rajoutant son adresse et ses intervalles d'associativité au paquet de la requête. Une fois que cette requête est reçue par la destination, cette dernière choisit le chemin avec le meilleur degré d'associativité. Elle envoie un paquet de réponse appelée **REPLY** au nœud source en utilisant le chemin choisi.

En plus des deux catégories de protocoles de routage, proactifs et réactifs, il existe une autre catégorie appelée hybride qui combine les précédentes. Cette catégorie fera l'objet du paragraphe suivant.

### 1.3.2.3 Les protocoles de routage hybrides

Ces protocoles utilisent l'approche proactive pour déterminer le voisinage à deux sauts ou trois sauts. De cette manière, les routes dans le voisinage sont définies. Au-delà de cette zone prédéfinie, les protocoles hybrides utilisent l'approche réactive pour la recherche de route. De ce fait, le réseau est découpé en plusieurs zones. À la réception d'une requête de recherche de route réactive, le nœud peut signaler si la destination est dans son voisinage ou non pour renvoyer la requête vers les autres zones. Le protocole de routage **Zone Routing Protocol (ZRP)** présente un exemple de cette catégorie.

Les protocoles hybrides s'adaptent bien aux grands réseaux, cependant, ils comportent aussi les inconvénients des protocoles réactifs tels que : les messages de contrôle périodiques, plus le coût de recherche d'une nouvelle route.

#### Exemple de protocole de routage hybride

Dans ce paragraphe, nous présentons le protocole de routage hybride ZRP :

- Le protocole de routage ZRP découpe le réseau en zones. Chaque zone du réseau inclut un ensemble de nœuds dont le nombre de sauts ne dépasse pas un certain nombre prédéfini appelé aussi rayon de la zone. Des messages de mise à jour sont échangés localement dans chaque zone afin de permettre aux nœuds de connaître localement la topologie du réseau.

TABLE 1.1 – Description des critères adaptés dans la classification des protocoles de routage MANET

Type	Description
Uniforme	tous les nœuds du nœud agissent de la même manière pour la fonction de routage : aucune hiérarchie dans le réseau
Non uniforme	quelques nœuds participent dans la fonction du routage : structure hiérarchique du réseau
Protocoles orientés topologie	chaque nœud collecte des informations sur l'état de ses connexions avec ses voisins et les transmet aux autres nœuds pour leur garantir une connaissance précise de la topologie du réseau
Protocoles orientés destination	chaque nœud échange ses estimations de distance avec chacun de ses voisins directs pour tous les nœuds de réseau
Les protocoles à partitionnement	le réseau est découpé en zones dans lesquelles le routage est assuré par un unique nœud maître
Les protocoles à sélection de voisins	chaque nœud traite la fonction de routage à un sous-ensemble de ses voisins directs

Dans une zone, un protocole de routage proactif tel que DSDV est utilisé pour se renseigner sur la topologie. Pour déterminer un chemin vers un nœud d'une zone extérieure, un protocole de routage réactif tel que DSR est employé.

Lors de la découverte de route dans ZRP, la source vérifie d'abord que la destination est hors de sa zone. Elle envoie alors un paquet de requête à tous les nœuds sur la périphérie de sa zone. Chacun des nœuds qui reçoit le paquet de requête lui ajoute son adresse et le transmet à ses nœuds périphériques, dans le cas où la destination n'est pas dans leur zone de routage. Dans le cas contraire, cas où un nœud identifie la destination comme membre de sa zone, le chef de cette zone envoie un paquet de réponse qui inclut la route complète vers la destination.

#### 1.3.2.4 Classification

Les protocoles de routage pour les réseaux ad hoc peuvent être classés de différentes manières. Le tableau 1.1 décrit quelques critères adaptés dans cette classification. Elle peut dépendre de :

- **le principe de conception** : proactif, réactif, ou hybride.
- **la structure du réseau** : qui peut être uniforme ou non uniforme.
- **l'état des informations** : obtenues par chaque nœud dans le réseau comme pour les protocoles de routage orientés topologie ou les protocoles de routage orientés destination.

Sur la base des critères décrits dans 1.1, la figure 1.7 présente une classification des protocoles de routage MANET [17], [37].

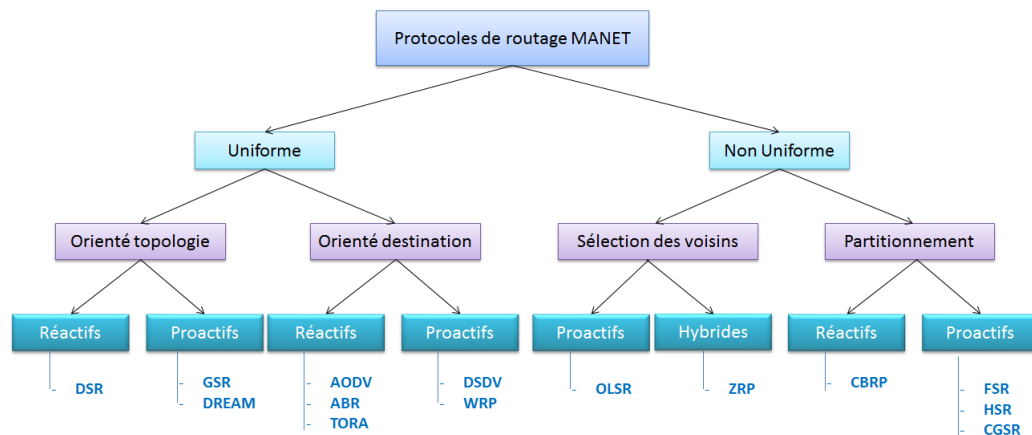


FIGURE 1.7 – Classification des protocoles de routage MANET

Dans ce qui suit, nous détaillons le fonctionnement du protocole de routage OLSR afin de justifier le choix de ce protocole dans notre étude.

## 1.4 Le protocole de routage OLSR

Le protocole OLSR est une optimisation de l'algorithme à état de liens. Le concept fondamental utilisé dans ce protocole est l'utilisation des relais multipoints (**MultiPoint Relays (MPR)**). Chaque nœud choisit dans son voisinage un ensemble des nœuds MPR. L'ensemble des nœuds MPR sont choisis de sorte qu'il couvre tous les nœuds qui sont à deux sauts. Les MPR expédient des messages de diffusion pendant le processus d'inondation et ils sont les seuls à déclarer leurs liens. Cette technique réduit considérablement la surcharge du réseau par rapport à un mécanisme classique d'inondation (où chaque nœud retransmet chaque message).

Le protocole de routage OLSR est conçu pour calculer et recalculer les routes. Ce protocole offre des routes optimales avec le nombre minimal de sauts. Dans le processus de calcul de route, il est indispensable de connaître les éléments suivants :

- la topologie du réseau : OLSR utilise une table topologique du réseau grâce aux messages de contrôle de topologie (**Topology Control (TC)**),
- la liste des voisins : les relations de voisinage à un saut et à deux sauts, ces relations étant définies par l'échange des messages HELLO,
- leurs types de liens : lien symétrique (si la communication est possible dans les deux sens), lien asymétrique (si la communication n'est possible que dans un seul sens), perdu (si le lien a été perdu) ou non spécifié.

Chaque nœud du réseau maintient une table de routage qui lui permet de router les paquets de données, à destination des autres nœuds du réseau. Le calcul de la table de routage est basé sur



les informations locales de chaque nœud et les informations de l'ensemble de la topologie. Par conséquent, la modification de l'une de ces informations entraîne le recalcul de la table de routage pour mettre à jour les informations de route pour chaque destination du réseau. Chaque entrée de la table de routage se compose de l'adresse du nœud destinataire ( $R\_dest\_addr$ ), l'adresse de premier saut à suivre ( $R\_next\_addr$ ), le nombre de sauts qui sépare les deux nœuds ( $R\_dist$ ), ainsi que l'interface de nœud local ( $R\_iface\_addr$ ). Cette entrée indique que le nœud destination identifié par  $R\_dest\_addr$  est estimé à  $R\_dist$  sauts du nœud local.  $R\_next\_addr$  est l'adresse du nœud suivant à un saut dans la route vers la destination  $R\_dest\_addr$ . Ce nœud voisin symétrique est accessible à travers l'interface locale avec l'adresse  $R\_iface\_addr$ . La table de routage contient des entrées pour chaque destination dans le réseau pour lequel il existe une route. Les destinations pour lequel une route est rompue ou partiellement connue, ne sont pas enregistrées dans la table de routage [28].

Chaque nœud accumule des informations sur le réseau, grâce à l'échange de messages de contrôle OLSR. Ces informations sont stockées dans différentes bases de données. Les deux principales fonctionnalités sont : la découverte des voisins (découverte de lien), et la diffusion des informations topologiques. Dans le paragraphe suivant, nous détaillons ces deux fonctionnalités.

### 1.4.1 Découverte des voisins

Chaque nœud du réseau diffuse périodiquement des messages 'HELLO' pour détecter ses voisins directs. Ces messages contiennent la liste des voisins connus du nœud et leur état de liens. L'état de la liaison peut-être soit symétrique, asymétrique, perdu ou relais multipoint. Ce dernier état signifie que le lien est symétrique et que l'expéditeur du message HELLO a choisi ce nœud comme un MPR. Les messages HELLO sont diffusés à chaque intervalle de temps ( $HELLO\_Interval$ ), fixé à 2 secondes par défaut. Ces HELLO sont reçus par tous les voisins à un saut, mais ils ne sont pas transmis. Ainsi, les messages HELLO permettent la découverte de voisinage à un saut, et à deux sauts pour chaque nœud. Un intervalle de maintien de validité est associé aux informations de voisinage à un saut et à deux sauts. Cet intervalle ( $Neighbor\_hold\_time$ ) est défini par défaut à  $3 * HELLO\_Interval$  [38].

La figure 1.8 montre un exemple de détection de voisinage par échange de messages HELLO :

1. à  $t_1$ , A envoie un HELLO vide à B qui enregistre A comme voisin asymétrique puisque l'adresse de B ne figure pas dans le message HELLO ;
2. à  $t_2$ , B envoie un HELLO à A en déclarant ce dernier comme son voisin symétrique en mettant son adresse dans le message ;
3. à  $t_3$ , A trouve sa propre adresse dans le HELLO envoyé par B, alors il envoie un HELLO à B en enregistrant l'adresse de B comme voisin symétrique ;

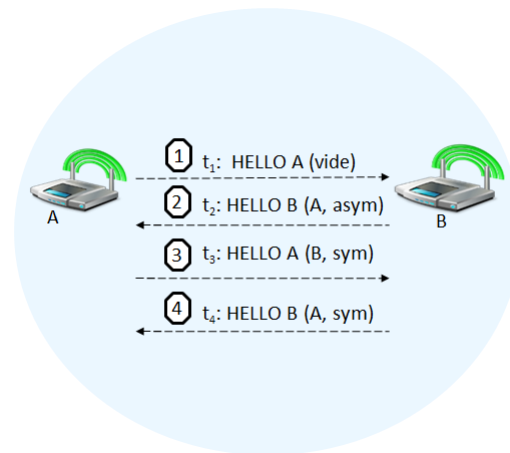


FIGURE 1.8 – Détection du voisinage par l'échange de messages HELLO

4. à  $t_4$ , de même B trouve son adresse dans le HELLO envoyé par A, alors il envoie un HELLO à B en enregistrant l'adresse de A comme voisin symétrique.

À partir de l'échange de messages HELLO, chaque nœud peut construire sa propre base de données appelé Link\_Tuple. Cette base décrit le voisinage d'un nœud à un saut en précisant le type de lien, le voisinage à deux sauts et la liste de ses MPR. La figure 1.9 illustre un exemple de topologie ad hoc et affiche la base Link\_Tuple des nœuds A et E, suivant les caractéristiques de cette topologie.

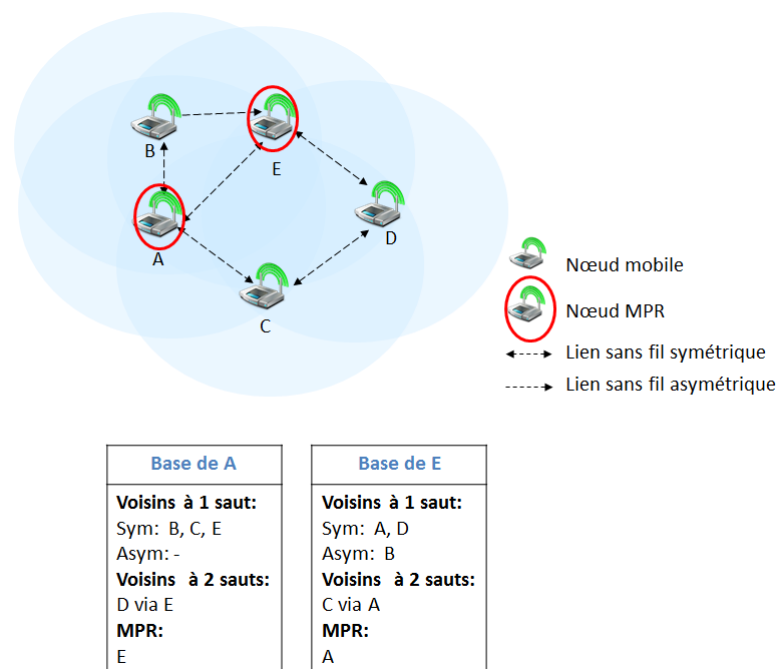


FIGURE 1.9 – La base Link\_Tuple d'un nœud dans OLSR

Sur la base de ces informations, les MPR sont sélectionnés pour chaque nœud à partir des voisins à un saut. Ils sont choisis de manière à couvrir tous les voisins à deux sauts. L'ensemble des MPR est recalculé à chaque fois qu'un changement de voisinage à un saut ou à deux sauts est

déecté. Lors de la phase de l'inondation, seul les MPR diffuse les messages en utilisant la règle suivante :

**Un nœud retransmet un message si et seulement si :**

- il ne l'avait pas déjà reçu,

**et**

- il vient de le recevoir d'un nœud dont il est MPR.

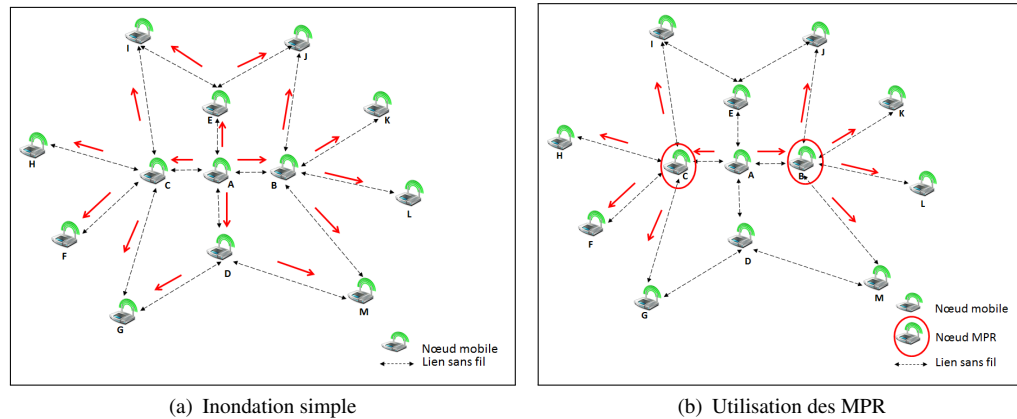


FIGURE 1.10 – Comparaison entre une inondation simple et l'utilisation des MPR

La figure 1.10 illustre une comparaison entre l'utilisation d'une inondation simple et l'utilisation des MPR dans une topologie ad hoc. Nous observons que les MPR présente un gain en nombre de retransmissions. Dans la figure 1.10 (a), tous les voisins d'un nœud retransmettent l'information, soient 12 répétitions. Par contre, la figure 1.10 (b), seulement les MPR retransmettent l'information il y a donc une économie de 4 retransmissions dans le réseau.

## 1.4.2 Découverte de la topologie

Chaque nœud du réseau utilise les messages TC pour maintenir ces informations topologiques sur le réseau. Les nœuds MPR diffusent un message TC à chaque TC\_Interval, défini par défaut à 5 secondes. Lors d'un changement de l'ensemble des MPR, le prochain message TC peut être envoyé plus tôt. Tous les nœuds du réseau diffusent les messages TC par inondation et les MPR réduisent le nombre de retransmissions. De cette manière, un nœud peut être atteint soit directement, soit par l'intermédiaire de ses MPR. Les informations de contrôle de topologie recueillies par chaque nœud ont aussi un temps de maintien de validité associé 'Top\_hold\_time'.

Les informations spécifiques à la découverte des voisins et à la découverte de la topologie sont mis à jour périodiquement. Ces informations permettent à chaque nœud de calculer des chemins vers toutes les destinations connues. Les chemins étant calculés selon l'algorithme Dijkstra, par

conséquent, ils sont optimaux en terme de nombre de sauts. Chaque fois qu’il y a un changement dans les informations de voisinage ou de la topologie, la table de routage est recalculée.

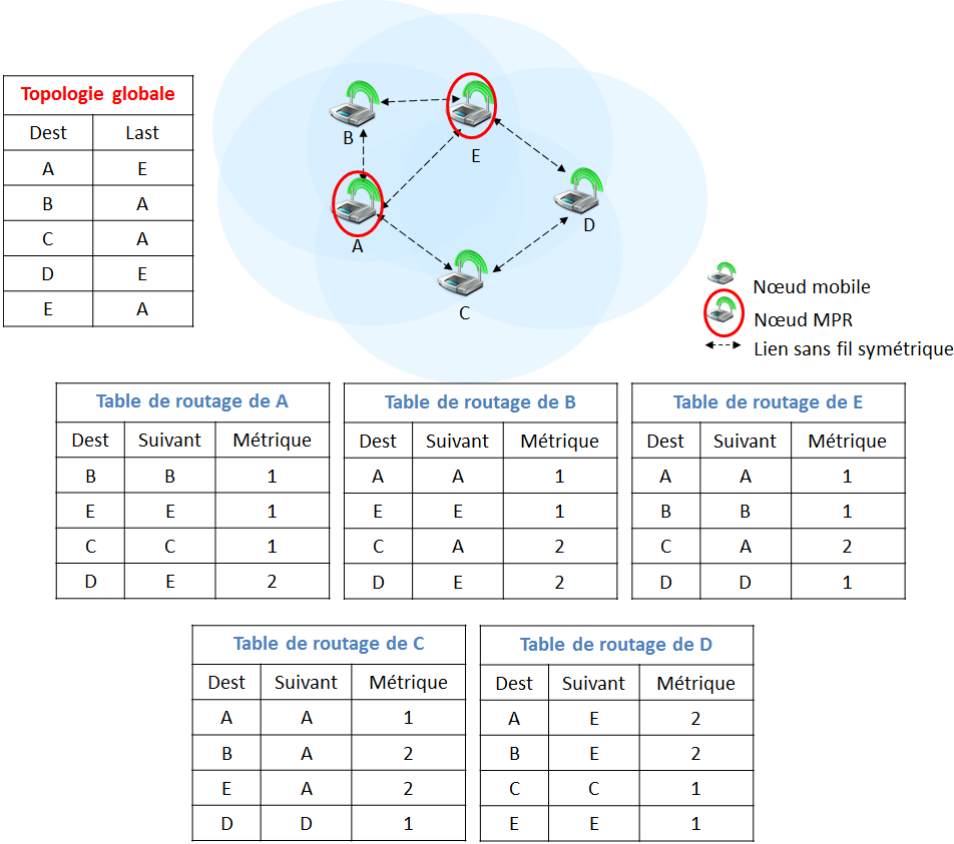


FIGURE 1.11 – Exemple de routage dans OLSR

Le routage dans OLSR se fait saut par saut. La figure 1.11 illustre un exemple de routage OLSR dans un petit réseau ad hoc. Chaque nœud du réseau calcule sa table de routage pour atteindre les autres nœuds, en se basant sur les informations de voisinage et de topologie mises à jour périodiquement. Dans la figure 1.11, nous observons que chaque nœud maintient sa propre table de routage. Cette table contient l’adresse de chaque nœud du réseau comme destination, le nœud suivant à utiliser pour atteindre la destination, et la métrique exprimée en nombre de sauts. À partir de ces informations, une table de topologie globale est construite qui donne une idée sur la topologie du réseau. Cette table contient les adresses de tous les nœuds comme destination et le nœud last qui précède la destination.

### 1.4.3 Choix du protocole de routage OLSR

Après une étude des différents protocoles de routage existants dans les réseaux ad hoc, nous sommes intéressés à choisir un. Notre choix s’est porté sur OLSR. Vue son comportement stable au niveau de la qualité des routes fournit mais aussi en délai de transmission, OLSR fait une optimisation des messages de routage surtout, quand le nombre de noeuds est important.

Le protocole OLSR maintient la mise à jour de ses tables de routage grâce à l'échange périodique de messages de contrôle tel que les messages TC et les messages HELLO. Ces messages de mises à jour peuvent refléter la qualité du lien entre deux nœuds. L'échange de messages HELLO donne une idée sur le taux de perte entre chaque paire de nœud, et l'échange de messages TC permet de propager cette information dans le réseau. Alors, selon ce principe de fonctionnement, le protocole OLSR est le mieux adapté dans notre étude dont le but est la proposition de nouvelles métriques de routage sensibles à la mobilité et qui reflètent la qualité du lien en même temps.

## 1.5 Conclusion

Dans ce chapitre, nous avons présenté les réseaux mobiles ad hoc, leurs principales caractéristiques, et leurs applications. De nombreux modèles de mobilité ont été détaillés afin de représenter le déplacement de mobiles. Nous avons souligné le problème du routage dû à la mobilité. Un groupe de protocoles de routage est discuté afin de déduire une classification de ces protocoles. Enfin, nous avons mis l'accent sur le protocole de routage OLSR sur lequel se base notre étude.

Le protocole de routage assure la fonction du routage à l'aide d'une métrique de routage. Dans la littérature, plusieurs métriques ont été définies. Dans le chapitre suivant, nous allons décrire un ensemble des métriques les plus utilisées par les protocoles de routage dans les réseaux ad hoc. Les définitions de ces métriques vont nous amener par la suite à faire une classification, afin de raffiner notre choix dans les métriques à étudier dans notre approche.

## Chapitre 2

# Classification des métriques dans les réseaux ad hoc

### Sommaire

---

<b>2.1 Métriques de routage</b>	<b>31</b>
<b>2.2 Métrique basée sur la topologie du réseau</b>	<b>32</b>
2.2.1 La métrique Hop Count	33
<b>2.3 Métriques basées sur la mesure de la puissance du signal</b>	<b>34</b>
<b>2.4 Métriques de la qualité de lien</b>	<b>35</b>
2.4.1 La métrique ETX	35
2.4.2 La métrique ETT	37
2.4.3 La métrique MTM	38
<b>2.5 Métriques de charge du trafic</b>	<b>39</b>
2.5.1 La métrique EAB	39
2.5.2 La métrique MF-Transmission Failure	40
2.5.3 La métrique C2WB	41
2.5.4 La métrique MIND	42
<b>2.6 Les métriques multi-canaux</b>	<b>43</b>
2.6.1 La métrique WCETT	43
2.6.2 La métrique MIC	46
2.6.3 La métrique MCR	47
2.6.4 La métrique iAWARE	47
2.6.5 Les métriques mETX & ENT	48
2.6.6 La métrique DBETX	49
2.6.7 Les métriques BATD & iBATD	50
<b>2.7 Métriques sensibles à la mobilité</b>	<b>52</b>
2.7.1 Les métriques Link Associativity Ticks & Path Aaverage degree of association stability	52

2.7.2	Les métriques Link Affinity & Path Stability . . . . .	53
2.7.3	Les métriques Link Availability & Path Availability . . . . .	54
2.7.4	La métrique LD . . . . .	55
2.7.5	La métrique Link Change Rate, Link State Changes $\lambda_{lc}$ . . . . .	57
2.7.6	La métrique Link Stability $L_s$ . . . . .	57
2.8	<b>Taxonomie des métriques de routage</b> . . . . .	<b>58</b>
2.9	<b>Choix des métriques</b> . . . . .	<b>58</b>
2.10	<b>Conclusion</b> . . . . .	<b>59</b>

Le routage dans les réseaux ad hoc sans fil est un domaine de recherche actif depuis de nombreuses années. Une grande majorité des protocoles de routage proposés choisit des chemins qui minimisent le nombre de sauts (Hop Count) car c'est la métrique la plus utilisée dans les réseaux filaires. Cependant les réseaux sans fil doivent faire face à un environnement plus complexe dans lequel intervient d'autres facteurs comme les interférences ou la gestion d'énergie. Par conséquent, une variété de métriques a été proposée pour les réseaux ad hoc sans fil afin de fournir des algorithmes de routage ayant une grande flexibilité dans le choix de la meilleure route et offrant un compromis entre le délai de bout-en-bout, et le débit.

Les métriques de routage jouent un rôle essentiel dans la sélection de chemins et l'optimisation des routes dans les réseaux ad hoc. Elles sont composées généralement d'un ensemble de paramètres qui permettent de capturer quelques critères des différentes couches du modèle OSI pour prédire la qualité de la liaison telle que : le taux de pertes de paquets, le débit, le délai de bout-en-bout, les interférences. Sur la base de ces paramètres, plusieurs classifications des métriques telles que [39], [40], [41] ont été proposées dans la littérature. Ces travaux présentent une taxonomie qui peut être utilisée pour comprendre, classer et comparer les différentes métriques mais ils ne traitent pas la catégorie des métriques sensibles à la mobilité. Cette catégorie est détaillée dans [42] ainsi que la catégorie des métriques qui optimisent la consommation d'énergie, dans le cas des réseaux de capteurs sans fil.

Ce chapitre identifie et donne une définition complète des principales métriques de routage, leurs objectifs d'optimisation, et leur méthode de calcul. Certaines métriques sont traitées plus en détail, d'une part parce qu'elles sont les premières à être introduites dans la littérature, et d'autre part parce qu'elles ont donné naissance à de nombreuses variantes. Notre description est délibérément comparative, montrant les similitudes et les différences entre les différentes catégories et les avantages relatifs de chaque métrique. Nous terminons alors par une classification des différentes métriques.

## 2.1 Métriques de routage

De nombreux protocoles de routage utilisent des métriques pour déterminer le plus “court” chemin à partir d’une source vers une destination. Une métrique est une valeur numérique associée à chaque lien. Le plus “court” chemin représente le minimal coût de ce chemin vis-à-vis de cette métrique. Elle doit garantir l’isotonicité<sup>1</sup> [43], [44]. La métrique reflète généralement le coût d’utilisation d’une route particulière par rapport à un objectif d’optimisation [42]. Le tableau 2.1 présente quelques objectifs d’optimisation d’un algorithme de calcul de route ainsi que celui de la métrique de routage.

TABLE 2.1 – Objectifs d’optimisation d’une métrique de routage

<b>Maximiser la probabilité de transmission des données :</b> minimiser le taux de pertes des données dans le réseau.
<b>Minimiser le délai :</b> sélectionner le chemin assurant un délai minimum.
<b>Maximiser le débit d’un chemin :</b> sélectionner un chemin de bout-en-bout composé de liens de grande capacité.
<b>Répartir équitablement la charge de trafic :</b> équilibrer la charge de trafic de sorte qu’aucun nœud (ou lien) ne soit disproportionnellement utilisé.

Les informations brutes sur un lien sont acquises à partir des mesures passives ou des mesures actives. Elles nécessitent généralement un traitement avant d’être utilisées pour construire une métrique de lien efficace et stable. Il existe différentes méthodes pour que les nœuds du réseau obtiennent les informations dont ils ont besoin pour le calcul de la métrique de routage [42] :

- **Réutilisation des informations disponibles localement :** les informations requises par la métrique sont disponibles localement au niveau du nœud tel que : le nombre d’interfaces du nœud, le nombre de voisins.
- **Contrôle (Surveillance) passif :** l’observation du trafic entrant et sortant d’un nœud permet de recueillir des informations pour la métrique, qui peuvent être combinées à d’autres critères.
- **Sondage (Probing) actif :** des paquets spéciaux sont générés afin de mesurer les caractéristiques ou les propriétés d’un lien ou d’un chemin. La technique de sondage permet d’effectuer des mesures actives sur la qualité des liens à partir de paquets sondes. La taille des paquets sonde a été le sujet de nombreuses discussions [45, 46]. D’une part, la taille doit être égale à celles des paquets de données pour donner des mesures pertinentes. D’autres part, les paquets sondes influencent la qualité du trafic si elles sont entrelacées

1. Une métrique isotonique devrait veiller à ce que l’ordre des coûts (poids) de deux chemins est conservé si ils sont ajoutés ou précédés par un troisième chemin commun. L’isotonicité est la condition nécessaire et suffisante d’une métrique de routage pour l’existence d’algorithmes efficaces pour trouver un chemin avec un coût minimal et sans boucler à l’infini, comme les algorithmes Bellman-Ford ou Dijkstra.



avec le trafic régulier du réseau. Les paquets de sonde permettent de mesurer directement la qualité de la liaison désirée, ce qui est mieux que de se baser sur des hypothèses analytiques [42, 47]. Ce mécanisme entraîne cependant une grande surcharge (overhead) du réseau, qui est directement dépendante de la fréquence des mesures et donc de leur précision.

- **Sondage Piggyback** : dans cette méthode, les mesures sont effectuées par sondage actif. Mais les informations nécessaires sont rajoutées dans une en-tête spécifique aux paquets réguliers. Avec cette technique, aucun paquet supplémentaire n'est généré afin de calculer la métrique, réduisant ainsi la surcharge du réseau. Le sondage Piggyback est une méthode courante pour mesurer le délai.

Le protocole de routage prend la décision finale pour choisir la route sélectionnée parmi plusieurs routes (si elles existent) pour transmettre les données. Par conséquent, la valeur de la métrique finale de chaque route qui sera l'objet de la comparaison porte sur l'ensemble du parcours (chemin). Cependant, la métrique d'un chemin est calculée à partir d'une fonction qui permet de prédire la valeur des métriques estimées pour chaque lien formant le chemin en question [42]. Les fonctions les plus utilisées dans les mesures des métriques sont la somme, la multiplication et les mesures statistiques tel que minimum, maximum et moyenne. La somme des valeurs des métriques des liens formant un chemin donne la valeur de la métrique de ce chemin. Le délai est un exemple des métriques additives. La multiplication des valeurs estimées de chaque lien fournit la métrique globale d'un chemin. Le taux de livraison est un exemple de métriques multiplicatives. Il existe aussi des métriques de chemin qui coïncident avec le minimum, la moyenne, ou le maximum des valeurs des liens du trajet. Le débit d'un chemin est un exemple de métriques qui considèrent le débit minimal d'un lien formant un chemin.

Nous avons défini ce qu'étaient une métrique de routage. Ses objectifs d'optimisations, ainsi que sa méthode de calcul. Dans la littérature, il existe une grande variété de métriques de routage et nous allons essayé de définir les plus citées et les plus appropriées dans le contexte de cette thèse. Dans la section suivante, nous présentons notre propre classification des métriques de routage selon six catégories : métriques topologique, métriques de mesure de la puissance du signal, métriques de la qualité de lien, métriques d'équilibrage de la charge de trafic, métriques multi-canaux, et métriques sensibles à la mobilité.

## 2.2 Métrique basée sur la topologie du réseau

Dans cette technique, les métriques considèrent les informations liées à la topologie du réseau, en particulier le nombre de voisins pour chaque nœud et le nombre de sauts pour atteindre une destination particulière. Ces métriques utilisent juste les informations de connectivité entre les

nœuds sans prendre en compte d'informations sur la qualité de la liaison reflétant les performances du réseau, tel que le débit, le taux de pertes ou le délai. L'exemple le plus connu de ces métriques est la métrique Hop Count.

### 2.2.1 La métrique Hop Count

Dans les réseaux mobiles ad hoc, la métrique Hop Count (nombre de saut (2003)) [3] est la plus utilisée par la plupart des protocoles de routage. Elle est utilisée en particulier dans les protocoles définis par l'IETF [48], tels que : OLSR [28], AODV [30], DSR [49] et DSDV [20]. Hop Count indique seulement l'existence d'un lien entre deux nœuds. Elle est égale à 1 si le lien existe, et  $+\infty$  si le lien n'existe pas. Elle est simple à calculer et elle évite toute charge de calcul supplémentaire pour le protocole de routage.

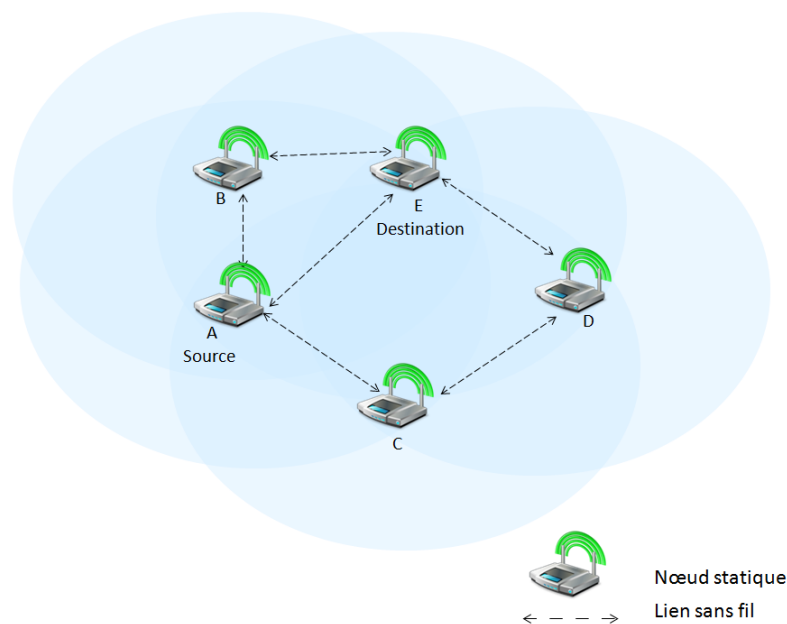


FIGURE 2.1 – Exemple de topologie de réseaux ad hoc : 5 nœuds statiques liés par des liaisons sans fil ; le flux de données circule de la source A vers la destination E

La figure 2.1 illustre un exemple simple d'une topologie de réseau ad hoc. La métrique Hop Count choisit le chemin avec le nombre minimum de sauts de la source A pour atteindre la destination E. Le tableau 2.2 montre le chemin approprié et le nombre de sauts nécessaires pour la communication de la source A avec le reste des nœuds du réseau, en appliquant la métrique Hop Count. La route sélectionnée par Hop Count de la source A vers la destination E est la route directe  $A \rightarrow E$  car E est un voisin direct de A.

Cette métrique favorise les chemins avec un petit nombre de sauts mais elle tend aussi à privilégier les liaisons à longue distance qui peuvent offrir des faibles puissances du signal reçues et des taux de pertes de paquets élevés. En outre, les liaisons à longue distance sont plus sensibles

TABLE 2.2 – Table de routage du nœud A avec la métrique Hop Count

Nœud	Route	Nombre de saut
B	$A \rightarrow B$	1
C	$A \rightarrow C$	1
D	$A \rightarrow C \rightarrow D$ ou $A \rightarrow E \rightarrow D$	2
E	$A \rightarrow E$	1

à la mobilité et aux ruptures de lien, en particulier pour les nœuds qui sont à la limite de leur portée radio les uns des autres. Par contre, les auteurs de [50] indiquent que Hop Count peut être intéressante dans le cas où aucun lien de qualité n’a pu être trouvé, en cas de mobilité. Dans nos simulations, nous montrons que Hop Count est très sensible à la mobilité.

Hop Count peut être inapproprié dans les réseaux sans fil car elle ne prend pas en compte la qualité de la liaison [50, 51] tel que le taux de pertes, la puissance du signal reçue, le débit, etc. Afin de remédier à ce problème, plusieurs métriques plus riches ont été proposées dans la littérature [52], [53].

### 2.3 Métriques basées sur la mesure de la puissance du signal

La mesure de la puissance du signal reçue a été considérée comme une métrique de routage dans les réseaux mesh sans fil (2009) [42]. La puissance du signal peut être utilisée comme un bon indicateur de la qualité du lien, avec l’hypothèse qu’un paquet de données est bien reçu si la puissance du signal reçue dépasse un certain seuil. Aujourd’hui, certaines cartes de réseau sans fil donnent une valeur de la puissance du signal reçue pour chaque paquet reçu avec succès.

Les auteurs de [54] (1999) considèrent la puissance du signal comme un facteur de qualité de lien. Chaque lien se voit affecté par la mesure de la puissance du signal comme coût du lien. Le facteur de qualité de lien est le produit des probabilités calculées pour chaque saut. Pour une route de M sauts, le facteur de qualité de lien L de cette route est estimé comme suit :

$$L = \prod_{s=1}^M (1 - Q((P_{pred_i} - P_{th})/\sigma)) \quad (.2.3.1)$$

où  $Q(x)$  est la fonction standard d’erreur  $Q$  – fonction,  $P_{pred}$  est la prédiction théorique de la puissance du signal reçue entre le nœud  $i$  et le nœud  $(i - 1)$ .  $P_{th}$  est le seuil de réception, et  $\sigma$  est la variance du signal, qui sont supposées être distribuées selon une loi normale.

Les auteurs de [55] (2001) mesurent la puissance du signal lorsqu’un paquet est reçu (contrôle passif). Ils définissent une région de préemption autour de la source, considérant que lorsque

la puissance du signal du paquet reçue est inférieure à un seuil prédéfini préventif  $P_{threshold}$ , le chemin d'accès est susceptible de se briser. Le seuil est défini comme suit :

$$P_{threshold} = \frac{P_0}{r_{preemptive}^4} \quad (.2.3.2)$$

où  $P_0$  est une constante qui dépend des caractéristiques des antennes, définie pour chaque paire émetteur et récepteur.  $r_{preemptive}$  est le rayon de la région de préemption du nœud.

A la réception d'un paquet, chaque nœud génère un message d'avertissement pour la source dès que la puissance du signal d'un paquet reçu franchit le seuil  $P_{threshold}$ . Alors la source cherche une autre route avec une meilleure qualité. Ce mécanisme présente un inconvénient majeur du fait qu'il ne considère que les liens symétriques. Néanmoins, l'évaluation des performances de cette méthode montre une amélioration significative des deux protocoles de routage, AODV et DSR. Cette méthode réduit bien le nombre des chemins qui vont se briser.

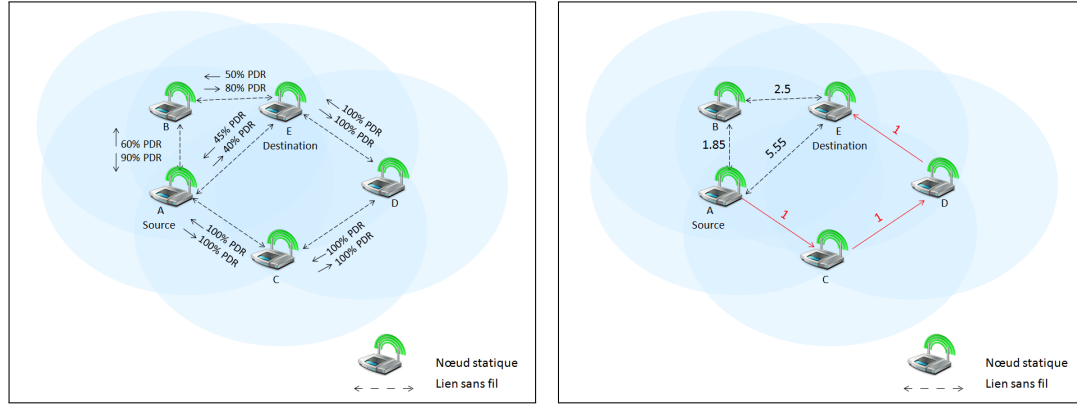
Une autre approche similaire nommée **Signal Stability based Adaptive Routing (SSAR)** a été proposée [56] (1997). Cette approche utilise des balises périodiques de la couche liaison afin d'avoir une estimation sur la qualité de la liaison. Seules les routes avec des liens stables et une bonne puissance de signal diffusent les paquets de découverte de route. La décision d'éliminer des routes dans cette méthode est prise par la source. Par contre dans la méthode de routage avec préemption, la décision est plutôt distribuée.

## 2.4 Métriques de la qualité de lien

Pour améliorer le routage, d'autres paramètres de qualité de lien ont été considérés pour trouver une meilleure route, tels que le taux de pertes, la bande passante ou le délai d'un chemin.

### 2.4.1 La métrique ETX

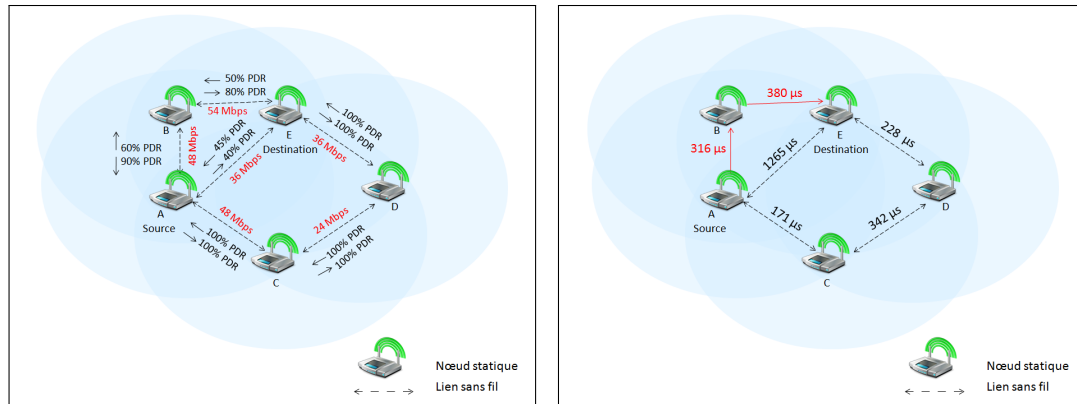
Afin de prendre en compte la qualité du lien, des métriques plus complexes ont été définies comme **Expected Transmission Count (ETX)** [4] (2003). Cette métrique est calculée en fonction du taux de pertes de paquet pour chaque lien, et pour chaque nœud. Le calcul d'ETX est basé sur la probabilité qu'un paquet de données est transmis avec succès entre une source et une destination d'une manière bidirectionnelle. Elle utilise le "*forward*" et le "*reverse delivery ratio*" d'un lien. Le "*forward delivery ratio*", **df**, est la probabilité qu'un paquet de données atteint la destination avec succès. Le "*reverse delivery ratio*", **dr**, est la probabilité qu'un paquet de données est reçu avec succès. **dr** est calculé à partir de la réception d'un **ACKnowledgement packet (ACK)** que la source envoie à la destination pour l'informer qu'elle a bien reçu le paquet.



(a) Exemple de topologie de réseau ad hoc en illustrant le taux de pertes pour chaque lien sans fil

(b) Route sélectionnée en utilisant la métrique ETX

FIGURE 2.2 – Métrique ETX dans un exemple de topologie de réseau ad hoc



(a) Topologie de Réseaux ad hoc en illustrant le débit et le taux de pertes pour chaque lien sans fil

(b) Route sélectionnée en utilisant la métrique ETT

FIGURE 2.3 – Métrique ETT dans un exemple de topologie de réseau ad hoc

La probabilité qu'une transmission soit reçue avec succès est définie par  $df * dr$ . ETX est défini comme suit :

$$ETX_l = \frac{1}{df * dr} = \frac{1}{fiability(l)} \quad (.2.4.3)$$

Par conséquent, ETX estime le nombre moyen de transmissions (y compris les retransmissions) nécessaires à la réception d'un paquet avec succès pour un lien  $l$ . L'utilisation de cette métrique pour calculer un chemin, favorise les liens les plus fonctionnels et les liens à haute capacité.

La figure 2.2(a) illustre la même topologie de réseau ad hoc que nous avons utilisée pour la métrique Hop Count, à laquelle on ajoute le taux de pertes sur chaque lien sans fil. Pour avoir

la valeur de la métrique ETX pour chaque lien sans fil, il suffit d'appliquer la formule 2.4.3. Prenons comme exemple le lien entre A et B,  $ETX_{AB}$  est calculée comme suit :

$$ETX_{AB} = \frac{1}{0.6 * 0.9} = 1.85 \quad (2.4.4)$$

La valeur de la métrique ETX pour chaque lien sans fil de la topologie est illustrée dans la figure 2.2(b). Il suffit d'additionner les valeurs des métriques correspondant aux liens formant un chemin, pour avoir le coût de ce chemin. La meilleure route est la route avec le minimal coût.

TABLE 2.3 – Coût des chemins possibles de la source A vers la destination E avec la métrique ETX

Chemin	Coût
$A \rightarrow E$	5.55
$A \rightarrow B \rightarrow E$	4.35
$A \rightarrow C \rightarrow D \rightarrow E$	3

Le tableau 2.3 montre les coûts des différents chemins possibles de la source A vers la destination E. La route de coût minimal est  $A \rightarrow C \rightarrow D \rightarrow E$  (voir figure 2.2(b)).

Le calcul d'ETX est basé sur la technique de sondage actif avec envoi de paquets de sonde généralement de taille plus petite que les paquets de données. Cette technique de sondage a été proposée afin de pallier les inconvénients des métriques de topologie. La qualité d'un lien calculée par ETX à partir des paquets de sonde ne donne pas nécessairement les mêmes performances si elle est calculée à partir des paquets de données [57]. En plus, ETX a été conçue pour les environnements à canal unique et à radio unique. Cette métrique ne fait pas de distinction entre les liens de capacités différentes.

## 2.4.2 La métrique ETT

La métrique Expected Transmission Time (ETT) [5] (2004) a été proposée pour prendre en considération les liens avec différentes capacités dans un réseau. Le calcul d'ETT est basé sur la métrique ETX mais ETT prend en compte à la fois la taille des paquets de données notée  $S$  et le débit noté  $B$  :

$$ETT_l = ETX_l * \frac{S}{B} \quad (2.4.5)$$

ETT peut être définie comme le délai prévu de transmission d'un paquet de longueur  $S$ . Elle multiplie le nombre de tentatives pour envoyer un paquet par le temps de transmission du paquet

sur le lien  $l$  ( $\frac{S}{B}$ ). Dans la même topologie utilisée pour Hop Count et ETX, la figure 2.3(a) présente le débit de chaque lien sans fil. Pour calculer ETT, nous appliquons la formule 2.4.5 pour chaque lien sans fil. Par exemple, dans le cas du lien entre A et B, ETT est calculée comme suit :

$$ETT_{AB} = ETX_{AB} * \frac{S}{B} = 1.85 * \frac{8192}{48} = 316\mu s \quad (.2.4.6)$$

Avec  $S = 1024 \text{ octets} = 8192 \text{ bits}$ . La figure 2.3(b) montre les valeurs de la métrique ETT pour chaque lien sans fil. La somme des valeurs des métriques de chaque lien formant un chemin présente le coût de ce lien. La meilleure route avec le minimal coût de A à E est  $A \rightarrow B \rightarrow E$ , représentée par la figure 2.3(b).

### 2.4.3 La métrique MTM

Une autre métrique similaire à ETT, **Medium Time Metric (MTM)** a été proposée par [58] (2004). Cette métrique est conçue pour les réseaux à débits multiples (multi rate networks). Dans ce type de réseau, les liens à longue distance ont une faible efficacité de débit. La métrique MTM minimise l'utilisation du support physique. Pour un paquet  $p$  dans un chemin  $C$ , la métrique MTM est définie comme suit :

$$MTM(C, p) = \sum_{\forall l \in C} \tau(l, p) \quad (.2.4.7)$$

où  $\tau(l, p)$  est le temps requis pour la transmission du paquet.  $\tau(l, p)$  est définie comme suit :

$$\tau(l, p) = \frac{\text{overhead}(l) + \frac{\text{size}(p)}{\text{rate}(l)}}{\text{fiability}(l)} \quad (.2.4.8)$$

L'overhead de la liaison comprend les trames de contrôle, le back-off, et les en-têtes fixes des trames.  $\text{Size}(p)$  est la taille du paquet  $p$ , et  $\text{rate}(l)$  est le débit de transmission du lien  $l$ .  $\text{fiability}(l)$  est définie comme la fraction de paquets transmis avec succès sur le lien  $l$ .

L'overhead d'un lien peut être calculé à partir de normes et de spécifications ainsi que le type et la configuration de la carte sans fil utilisée. La couche MAC fournit des informations sur la fiabilité et le débit de transmission de la liaison, mais ces deux dernières informations ne sont pas toujours accessibles à la couche Réseau (couche supérieure) car la couche MAC utilise des techniques de sélection de débit considérées comme exclusives. Pour remédier à ce problème, le sondage par sonde permet d'estimer la fiabilité et le débit de transmission de la liaison. Mais le sondage produit une surcharge du réseau avec des informations moins précises que la communication inter-couches.

Pour valider les performance de leur métrique, les auteurs de [58] mesurent le débit de bout-en-bout. Sur des courtes distances, les résultats de MTM ont donné les mêmes performances que ETX et Hop Count. Par contre avec des distances plus grandes, le débit de bout-en-bout avec la métrique MTM est 20 fois plus élevée qu'avec ETX et Hop Count. MTM choisit des chemins avec plus de sauts que ETX et Hop Count mais qui offre plus de capacité que les autres.

Nous avons présenté dans cette section quelques métriques qui reflètent la qualité du lien. Plusieurs études ont été effectuées dans le but de proposer des métriques qui prennent en compte d'autres contraintes de l'environnement sans fil dans le routage tel que la sensibilisation de la charge du trafic. Ce critère a un effet significatif spécialement dans les applications temps réel. Dans le paragraphe suivant, nous présentons quelques métriques permettant de capturer les paramètres de la concentration et de la congestion du trafic au niveau des nœuds.

## 2.5 Métriques de charge du trafic

Ces métriques choisissent les meilleures routes en fonction de l'estimation de la charge du trafic des nœuds formant le chemin, tandis que les métriques de qualité de lien choisissent leurs routes en fonction de la qualité des liens des routes.

### 2.5.1 La métrique EAB

La métrique **Expected Available & Bandwidth (EAB)** [59] (2010) a été proposée pour résoudre le problème des zones de concentration de trafic du réseau, en assurant un débit élevé et un faible délai moyen de bout-en-bout. EAB prend en compte la bande passante disponible et le taux de transmission réussie. La bande passante disponible est calculée comme estimation de la bande passante totale moins la largeur de la bande passante occupée par chaque lien sur un nœud. Si un lien a beaucoup de bande passante disponible, un nœud peut transmettre plus de quantité de données via ce lien. Cette métrique choisit un chemin qui a un faible délai de bout-en-bout et un taux de livraison élevé. Pour un lien  $l$  et à un instant  $t$ ,  $EAB(l, t)$  est exprimé comme :

$$EAB(l, t) = AB(l, t) + P_{success}(l, t) \quad (.2.5.9)$$

Le premier terme  $AB(l, t)$  est utilisé pour calculer la bande passante disponible sur le lien  $l$ , à un certain temps  $t$ .  $AB(l, t)$  est donnée par :

$$AB(l, t) = BW_{total}(l, t) - BW_{occupied}(l, t) \quad (.2.5.10)$$

$BW_{total}(l, t)$  et  $BW_{occupied}(l, t)$  sont respectivement la largeur de bande passante totale attribuée à un lien individuel  $l$  et la largeur de la bande passante occupée par chaque lien  $l$ . Le deuxième



terme dans  $EAB(l, t)$ ,  $P_{success}(l, t)$ , est la valeur de probabilité de transmission réussie de chaque liaison à l'instant  $t$  en utilisant le  $d_f$  et le  $d_r$  tels qu'ils sont décrits dans la section 2.4.1.  $P_{success}(l, t)$  est donnée par :

$$P_{success}(l, t) = d_f(l, t) * d_r(l, t) \quad (.2.5.11)$$

Dans [59], les auteurs ont comparé les performances de EAB avec d'autres métriques de routage tels que Hop Count, ETX et ETT. Les mesures prises en compte sont le débit et le délai moyen de bout-en-bout entre un nœud source et un nœud destination en fonction de l'augmentation de la charge de trafic. Les résultats de simulation montrent que EAB présente des meilleures performances que les autres métriques de routage pour la simple raison qu'elle sélectionne un chemin avec beaucoup de bande passante disponible et un taux de transmission de succès élevé. Par conséquent, EAB peut éviter les nœuds dans les zones à fort trafic, dans un réseau mesh par exemple.

## 2.5.2 La métrique MF-Transmission Failure

Les auteurs de [60] (2011) ont définie une nouvelle métrique **MF-Transmission Failure (MF)** dans le but de prendre en compte les échecs de transmission en considérant le mécanisme de backoff utilisé dans les réseaux IEEE 802.11 et en attribuant des coefficients à chaque chemin. Ces coefficients agissent comme métrique pour sélectionner une route entre différents chemins et pour assurer aussi l'équilibrage de charge. Par conséquent, cette métrique aide les protocoles de routage à équilibrer le trafic et à éviter le trafic à travers les chemins congestionnés. De plus, la métrique MF prend en considération l'interférence inter-flux et intra-flux<sup>2</sup>. MF est définie comme étant :

$$MF = x * (B(j)) + y * (C(j)) \quad (.2.5.12)$$

Les variables  $x$  et  $y$  désignent des pondérations, qui dépendent de l'application. Dans l'équation 2.5.12,  $x$  correspond au degré de fiabilité et  $y$  correspond à l'accomplissement de la demande de l'utilisateur. MF assure une sélection de chemin sur la base des échecs de transmission et de la capacité de nœuds pour chaque paire source/destination.  $B(j)$  présente la capacité résiduelle du nœud qui est calculée et annoncée comme la capacité de transmission de lien.  $C(j)$  sélectionne le chemin qui renvoie tous les nœuds dont la capacité résiduelle maximale.

Une analyse comparative entre la métrique MF-Transmission Failure et d'autres métriques de routage tel que Hop Count, ETX et ETT a été effectuée dans [60]. Elle mesure les performances

2. Dans un réseau sans fil, le canal de transmission est partagé par plusieurs liens, par conséquent, la qualité et la quantité des informations correctement reçues sont réduites. Si, par exemple, deux nœuds  $x$  et  $y$  transmettent des données au nœud  $z$  en même temps, le nœud  $z$  ne peut pas déchiffrer et comprendre correctement l'information envoyée. Ce phénomène est appelé interférence. Un nouveau flux est sujet à des interférences inter-flux engendrées par les flux existants et des interférences intra-flux causées par le flux lui-même.

des différentes métriques en termes de taux de pertes des paquets, de débit et de l'utilisation de la file d'attente des différents routeurs mesh de l'ensemble du réseau. Les résultats de simulation montrent que la métrique proposée est meilleure que Hop Count, ETX et ETT car elle permet de fournir un schéma d'équilibrage de charge dans le réseau.

### 2.5.3 La métrique C2WB

La métrique **Channel Utilization & Contention Window Based metric (C2WB)** [61] (2011) a été définie dans le but de résoudre le problème de l'interférence et d'améliorer la charge de trafic. Elle capture les interférences inter-flux et la congestion dans les réseaux mesh. C2WB prend en compte le temps de service de trame pour chaque liaison  $l$  sur un chemin  $p$ . Le nœud  $n$  est le nœud émetteur en considérant le lien  $l$ . C2WB est définie comme étant :

$$C2WB = \sum_{l \in P} T_{ser,l} = \sum_{l \in P, n \in P} \frac{1}{1 - C_n} \left[ (\overline{CW}_l - \frac{CW_0}{2}) T_{slot} + \frac{1}{1 - FER} \frac{PL}{B_e} \right] \quad (.2.5.13)$$

$T_{ser,l}$  est le délai de service sur le lien  $l$ . Ce terme est la somme du délai de back-off, le délai de report (deferring time) et le délai de transmission :

$$T_{ser,l} = T_{bac,l} + T_{def,l} + T_{trans,l} \quad (.2.5.14)$$

Le temps du back-off est exprimé en fonction de la fenêtre de contention moyenne (Average Contention Window) comme suit :

$$T_{bac,l} = (\overline{CW}_l - \frac{CW_0}{2}) T_{slot} \quad (.2.5.15)$$

La fenêtre de contention moyenne,  $\overline{CW}_l$ , est exprimée en fonction du taux d'erreur trame (Frame Error Rate ( $FER_l$ )) du lien  $l$  et la valeur de la fenêtre de contention au stade 0 notée ( $CW_0$ ).  $\overline{CW}_l$  est définie comme étant :

$$\overline{CW}_l = \frac{\sum_{i=0}^n (FER_l^i (1 - FER_l) CW_i)}{\sum_{i=0}^n (FER_l^i (1 - FER_l))} \quad (.2.5.16)$$

Les auteurs de [61] ont utilisé la même technique que celle utilisée pour la mesure de ETT (voir Section 2.4.2), pour déterminer le taux d'erreur de trame (FER) d'un lien comme :

$$FER = 1 - d_f * d_r \quad (.2.5.17)$$

Le temps de report (deferring time),  $T_{def,l}$ , est exprimé comme étant :

$$T_{def,l} = \frac{C_n}{1 - C_n} (T_{bac,l} + T_{trans,l}) \quad (.2.5.18)$$

où  $C_n$  définit l'utilisation du canal au nœud  $n$ , afin de déterminer l'occupation d'un canal (comme la fraction de temps de canal dans lequel le canal est détecté occupé par le nœud  $n$ ).

Le temps de transmission  $T_{trans,l}$  est exprimé en utilisant la bande passante efficace noté  $B_e$  et la taille de la trame des données utiles (Payload frame) notée  $PL$ . La valeur de  $PL$  est réglée à 1500 octets.

$$T_{trans,l} = \frac{1}{1 - FER_l} \frac{PL}{B_e} \quad (.2.5.19)$$

Les performances de la métrique C2WB sont comparées aux performances des deux métriques ETX et ETT dans [61]. Les résultats montrent que C2WB est plus performant qu'ETX et ETT, en présence d'interférence.

### 2.5.4 La métrique MIND

La métrique **M**etric of **I**nterference and channel **D**iversity (**MIND**) [62] (2009) capture l'interférence et la charge courante sur la base d'une technique de contrôle passif, réduisant ainsi l'overhead qui est d'habitude causé par le mécanisme de sondage actif. MIND est définie comme suit :

$$MIND = \sum_{link i \in P}^n INTERLOAD_i + \sum_{node j \in P}^m CSC_j \quad (.2.5.20)$$

Dans cette formule le composant INTERLOAD capture l'interférence inter-flux et la charge d'un lien  $i$ . Le composant CSC capture l'interférence intra-flux (le même calcul que l'Equation .2.6.30).  $n$  désigne le nombre total de liaisons sans fil, et  $m$  indique le nombre total de nœuds sur le chemin  $p$ . Le composant INTERLOAD est défini comme suit :

$$INTERLOAD_i = ((1 - IR_j) \cdot \tau) \cdot CBT_j \quad (.2.5.21)$$

Avec  $0 \leq IR \leq 1$  et  $0 \leq CBT \leq 1$ .  $IR$  est le rapport d'interférences en fonction du **S**ignal-to-**I**nterference-plus-**N**oise-**R**atio (**SINR**) et **S**ignal-to-**N**oise-**R**atio (**SNR**). Pour un lien  $j$ , cette composante est calculée comme suit :

$$IR_j = \frac{SINR_j}{SNR_j} \quad (.2.5.22)$$

$IR$  prend en compte les interférences entre les liens à travers les valeurs de la puissance du signal mesurées à partir des cartes sans fil. Elle détermine les valeurs du  $SINR$  et  $SNR$  en utilisant un mécanisme de monitoring (surveillance) passif sans trafic supplémentaire, contrairement à d'autres métriques de routage qui utilisent les paquets de sonde. Dans le cas où il n'existe pas de voisins interférents (pas d'interférence inter-flux),  $SINR_j$  est égale au  $SNR_j$  et donc  $IR_j$  est 1. Dans ce cas, la qualité de la liaison pour le lien  $j$  est déterminée par la composante d'interférence

intra-flux. Une description plus détaillée sur les formules de SINR et SNR est présentée dans le paragraphe 2.6.4.

La composante *CBT* (Channel busy time) désigne le temps d'occupation du canal. Cette entité est également calculée passivement afin de prédire la charge, en utilisant l'équation suivante :

$$CBT_j = \frac{Total\ time - Idle\ time}{Total\ time} \quad (.2.5.23)$$

où *Total time* est le temps écoulé entre la première tentative d'émission d'un paquet et la réception de son acquittement. *Idle time* est le délai du Back Off et le délai dans lequel les nœuds radio détectent que le support d'accès est libre. Ainsi CBT est la mesure du temps écoulé pendant les états de transmission, de réception et d'occupation [40].

La métrique MIND considère les interférences inter-flux et intra-flux, en utilisant des modèles physiques et logiques d'interférence. L'inconvénient majeur de MIND est sa nature non isotonique qui rend sa mise en œuvre assez complexe. En plus, MIND ne prend pas en compte l'asymétrie des liens qui peut induire des erreurs dans les paramètres de mesure de qualité de canal. MIND peut aussi sous-estimer les interférences du canal car elle considère la période de Back Off comme le temps d'inactivité (Idle Time) [40].

## 2.6 Les métriques multi-canaux

Les nœuds multi-canaux ou multi-radio sont prometteurs pour l'amélioration de la capacité dans ce type de réseau [63]. Il est important de préciser que le terme multi-canaux désigne l'utilisation de plusieurs canaux ou fréquence, par contre le terme multi-radio désigne l'utilisation de plusieurs cartes ou interfaces radios. Jusqu'ici, nous avons considéré que les réseaux à radio-unique ou à canal-unique, où la capacité des nœuds relais est divisée par deux pour assurer la bidirection. Par contre avec un réseau multi-canaux, un nœud peut envoyer et recevoir des données en même temps. Le nœud peut émettre sur deux canaux simultanément, si le réseau utilise plus de spectres en radio-fréquence. En plus, les radios qui fonctionnent sur différentes bandes de fréquence ont des bandes passantes différentes, une portée radio différente, et des caractéristiques d'évanouissement différentes. Par exemple, la norme *IEEE802.11a* utilise *5GHz* et la norme *IEEE802.11b/g* utilise *2.4GHz*.

Dans cette section, nous nous intéressons aux métriques dédiées aux réseaux multi-canaux.

### 2.6.1 La métrique WCETT

Les auteurs de [64] suggèrent que la somme de la métrique ETT de tous les liens d'un chemin, ne prend pas en compte le fait que les liens concaténés interfèrent les uns avec les autres, si

elles utilisent le même canal. Plusieurs technologies sans fil tel que le 802.11 a/b/g offrent des canaux multiples qui ne se chevauchent pas. Les auteurs de [64] proposent une adaptation de la métrique ETT en tenant compte de l'utilisation des canaux multiples, appelée **Weighted Cumulative ETT (WCETT)** (2004).

$$WCETT = (1 - \beta) * \sum_{i=1}^n ETT_i + \beta * \max_{1 \leq j \leq k} X_j \quad (.2.6.24)$$

$$X_j = \sum_{i \text{ utilise canal } j} ETT_i; 1 \leq j \leq k \quad (.2.6.25)$$

Avec  $0 \leq \beta \leq 1$  étant un paramètre ajustable.  $X_j$  est la somme des temps de transmission sur tous les sauts sur le canal  $j$ , et  $k$  est le nombre total de canaux du système. Comme le débit du trajet total est dominé par le canal du goulot d'étranglement (bottleneck channel), qui a le plus grand  $X_j$ , [64] proposent d'utiliser une moyenne pondérée entre la valeur maximale et la somme de tous les ETT.

WCETT est vue comme un compromis entre le débit et le délai. Le premier terme peut être considéré comme une mesure de la latence de ce chemin. Le second terme peut être considéré comme une mesure de débit du chemin, car il représente l'impact du goulot d'étranglement du saut (bottleneck hops). La moyenne pondérée est une tentative de trouver un équilibre entre les deux. Aussi, le deuxième terme améliore la diversité du canal et aide à trouver des chemins avec moins d'interférences intra-flux. En plus, WCETT n'est pas isotonique à cause du terme  $X_j$  qui capture l'effet de la diversité des canaux.

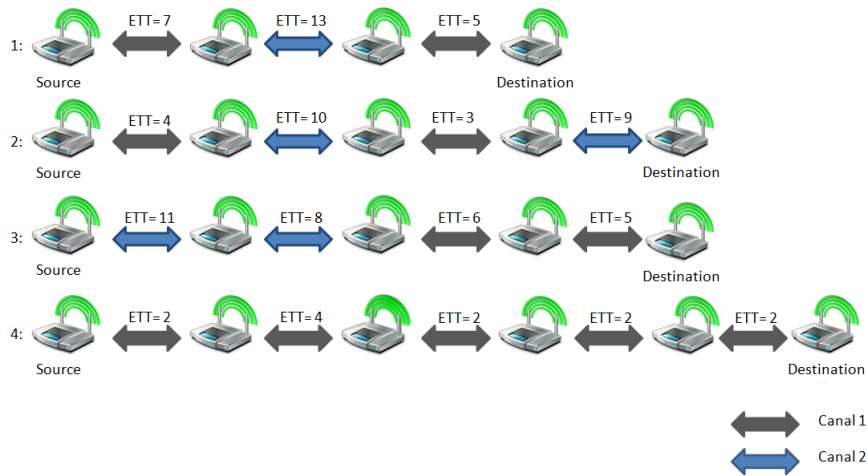


FIGURE 2.4 – Exemple de la métrique WCETT

La figure 2.4 présente un exemple d'un réseau où chaque nœud dispose de deux radios. Sur chaque nœud, une interface radio est allouée sur le canal 1 tandis que l'autre radio est assignée au canal 2. Nous avons supposé que les deux canaux n'interfèrent pas l'un avec l'autre et que

les deux interfaces possèdent des portées radio et des bandes passantes différentes. Comme le montre la figure 2.4, il existe quatre chemins possibles entre la source  $S$  et la destination  $D$ . Les valeurs de la métrique ETT sont présentées sur chaque lien de la figure. Le tableau 2.4 résume les étapes de calcul de la métrique WCETT pour  $\beta = 0.9$  et  $\beta = 0.1$ .

TABLE 2.4 – Calcul de la métrique WCETT

Chemin	Somme des ETT	Maximum des ETT	WCETT ( $\beta = 0.9$ )	WCETT ( $\beta = 0.1$ )
1	25	13	14.2	23.8
2	26	19	19.7	25.3
3	30	19	20.1	28.9
4	16	12	12.4	15.5

Considérant le premier chemin, nous observons que le goulot d'étranglement est le lien avec le canal 2. Le deuxième et le troisième chemin sont similaires au premier chemin, exception faite qu'ils comportent un saut de plus sur le canal 2. Toutefois, les liens sur le canal 2 présentent encore un goulot d'étranglement. Néanmoins, ces deux chemins sont en effet pires que le premier chemin, car ils incluent un saut supplémentaire. Ceci se reflète dans les calculs de WCETT dans le tableau 2.4. Toutefois, le chemin 2 reste meilleur que le chemin 3. Enfin, le chemin 4 bien qu'il n'ait pas une diversité du canal (il ne considère que le canal 1), est nettement meilleur que les trois premiers chemins. Cela se reflète correctement dans le calcul de la métrique WCETT dans le tableau 2.4.

WCETT a été mis en oeuvre sur un banc d'essai statique afin d'être comparé à ETX et Hop Count. Dans [64], il a été montré que WCETT est meilleure qu'ETX par un facteur de deux en terme de débit et meilleure aussi que Hop Count par un facteur de quatre, lorsque deux cartes radio IEEE 802.11 différentes ont été utilisées par station. WCETT présente un inconvénient principal qui se résume dans le fait qu'il n'existe pas (pour le moment) un algorithme qui peut calculer le chemin avec le poids le plus faible avec moins de temps.

Une nouvelle métrique **Modified Weighted Cumulative Consecutive ETT (ModWCCETT)** (2014) basée sur le calcul de WCETT est proposée dans [65]. ModWCCETT permet la sélection de chemin d'accès ayant le moins d'interférences et avec une plus grande disponibilité de canal dans un réseau maillé sans fil. Cette métrique propose une méthode pour mieux prendre en compte les interférences intra-flux, ModWCCETT peut être exprimée comme :

$$ModWCCETT_p = (1 - \beta) * \sum_{i=1}^n ETT_i + \beta * \max Z_j \quad (.2.6.26)$$

Avec :

$$Z_j = \sum_{\text{Conflicting link is on Channel } j} ETT_i \quad 0 \leq j \leq k \quad (.2.6.27)$$

Dans ModWCCETT, un segment est défini comme des sauts en conflit sur le même canal qui interfèrent les uns avec les autres. En conséquence, des segments différents ne peuvent pas interférer entre eux. ModWCCETT sélectionne un chemin avec une plus grande diversité de canaux (de plus petits segments) par rapport à **Weighted Cumulative Consecutive ETT (WCCETT)** [64] qui reflète moins l'interférence intra-flux.  $Z_j$  est la somme des valeurs ETT de liens qui sont sur le segment  $j$  dans un système. La comparaison des performances des métriques ModWCCETT, WCCETT, WCETT et ETT présentée dans [65], montre que ModWCCETT est meilleure que les autres métriques. ModWCCETT a un faible overhead dans le calcul de route. Aussi, l'interférence intra-flux est faible avec ModWCCETT en comparaison avec WCCETT, WCETT et ETT.

## 2.6.2 La métrique MIC

Les auteurs de [66] (2005) proposent la métrique **Metric of Interference and Channel-switching (MIC)** qui améliore la métrique WCETT en s'adressant aux problèmes de l'interférence inter-flux. Elle est définie comme suit :

$$MIC(p) = \frac{1}{N * \min(ETT)} \sum_{\text{lien } l \in p} IRU_l + \sum_{\text{nœud } i \in p} CSC_i \quad (.2.6.28)$$

où  $N$  est le nombre total de nœuds dans le réseau.  $\min(ETT)$  est la valeur minimale ETT estimée par la vitesse de transmission la plus basse des cartes sans fil. Le composant **Interference-aware Resource Usage (IRU)** gère l'interférence inter-flux, il correspond au temps total de canal consommé sur un lien  $l$ . La composante **Channel Switching Cost (CSC)** gère l'interférence intra-flux, en favorisant les chemins à canaux multiples et en pénalisant les chemins avec des liens consécutifs utilisant le même canal.  $IRU_l$  et  $CSC_i$  sont définis comme suit :

$$IRU_l = ETT_l * N_l \quad (.2.6.29)$$

$$CSC_i = \begin{cases} w_1 & \text{if } CH(\text{prev}(i)) \neq CH(i) \\ w_2 & \text{if } CH(\text{prev}(i)) = CH(i) \end{cases} \quad 0 \leq w_1 \leq w_2 \quad (.2.6.30)$$

Avec  $N_l$  est l'ensemble de voisins s'interférant avec lui sur la liaison  $l$ .  $CH(i)$  représente le canal de transmission attribué pour le nœud  $i$  et  $\text{prev}(i)$  représente le saut précédent du nœud  $i$  le long du chemin  $p$ . MIC n'est pas isotonique à cause du terme  $CSC_i$ , mais une décomposition assez complexe a été proposée par [66] pour rendre MIC isotonique.

La métrique MIC considère l'interférence intra-flux et inter-flux et la diversité des canaux, elle offre de meilleures performances que la métrique ETX. L'inconvénient de MIC est l'overhead important pour estimer la valeur de  $MIC(p)$  par chemin. En plus, chaque nœud doit connaître le nombre total des nœuds dans le réseau, ce qui semble très coûteux pour les grands réseaux [42].

### 2.6.3 La métrique MCR

La métrique Multi-Channel Routing (MCR) [67] (2006) est une nouvelle extension de WCETT mais de manière différente de MIC. MCR prend en compte le coût de changement de canal. Elle est définie comme suit :

$$MCR = (1 - \beta) * \sum_{i=1}^n (ETT_i + SC(c_i)) + \beta * \max_{1 \leq j \leq k} X_j \quad (.2.6.31)$$

Dans le premier composant, un coût de commutation est inclus dans la métrique ETT, afin d'empêcher la commutation fréquente de canal des chemins choisis.  $SC(c_i)$  est le coût d'utilisation du canal  $j$ , où *SwitchingDelay* représente le temps de latence de commutation de l'interface.  $p_s(j)$  est la probabilité que l'interface utilisée soit sur un canal différent quand on veut envoyer un paquet sur le canal  $j$ , où *InterfaceUsage(i)* est la fraction de temps qu'une interface commutable passe pour la transmission sur le canal  $i$ .

$$SC(c_i) = p_s(j) * SwitchingDelay \quad (.2.6.32)$$

$$p_s(j) = \sum_{\forall i \neq j} InterfaceUsage(i) \quad (.2.6.33)$$

Pour évaluer MCR, les résultats de simulation ont montré que la capacité du réseau peut être améliorée en utilisant de multiples canaux, même si seulement deux interfaces sont disponibles par nœud.

### 2.6.4 La métrique iAWARE

La métrique **interférence AWARE (iAWARE)** [68] (2006) estime le temps moyen pendant lequel le médium physique est occupé par la transmission de chaque voisin interférant. Cette métrique prend en compte les interférences inter-flow et intra-flow, mais aussi elle est caractérisée par un modèle physique d'interférence et elle considère aussi la variation de la qualité du lien. Afin de reproduire les variations des interférences entre les voisins, le calcul de iAWARE se base sur le rapport signal à bruit (Signal to Noise Ratio : SNR) et le rapport signal à bruit et interférence (Signal to Interference and Noise Ratio : SINR). Dans le modèle de [68], une communication entre un nœud  $u$  et un nœud  $v$  et représentée par le lien  $(u \rightarrow v)$ , est considérée



comme réussie si la valeur du SNR du nœud récepteur (le nœud  $v$ ) ne dépasse pas une certaine valeur de seuil. Les auteurs de [68] ont défini pour un nœud  $u$  et le lien ( $u \rightarrow v$ ), la variable Interference Ratio  $IR_i(u)$  ( $0 \leq IR_i(u) \leq 1$ ) définie comme suit :

$$IR_i(u) = \frac{SINR_i(u)}{SNR_i(u)} \quad (.2.6.34)$$

avec :

$$SNR_i(u) = \frac{P_u(v)}{N} \quad (.2.6.35)$$

$$SINR_i(u) = \frac{P_u(v)}{N + \sum_{w \in \eta(u)-v} \tau(w) P_u(w)} \quad (.2.6.36)$$

$P_u(v)$  est la puissance du signal du paquet du nœud  $u$  vers le nœud  $v$ .  $\eta(u)$  désigne l'ensemble des nœuds pour lesquels le nœud  $u$  peut détecter une transmission.  $\tau(w)$  est le taux normalisé au cours duquel le nœud  $w$  génère un trafic en moyenne sur une période de temps. Lorsque le nœud  $w$  envoie les paquets de données à son maximum du débit de données supporté,  $\tau(w)$  est égal à 1. [68] utilise  $\tau(w)$  pour pondérer la puissance du signal d'un nœud interférant  $w$ , puisque  $\tau(w)$  exprime la fraction de temps d'occupation du canal pour le nœud  $w$ .

La métrique iAWARE d'un lien  $j$  est exprimée comme suit :

$$iAWARE_j = \frac{ETT_j}{IR_j} \quad (.2.6.37)$$

En absence d'interférences,  $IR_j$  vaut 1, et  $iAWARE_j$  est égale tout simplement à  $ETT_j$  qui reflète le taux de pertes du lien et aussi le débit de transmission (transmission rate) du lien  $j$ .  $ETT_j$  est pondérée par  $IR_j$  pour capturer l'interférence subite par le lien de la part de ses voisins. La composante IR utilisée dans le calcul de MIND .2.5.20 est assez différente de la composante IR conçue pour la métrique iAWARE. En effet, le principe de conception de IR dans MIND est basé sur les paramètres de nœuds alors que la conception de IR dans iAWARE est basée sur les paramètres de liaison.

Si un lien a une faible valeur de  $ETT$  et une valeur élevée pour  $IR$ , iAWARE aura une faible valeur et inversement. Un lien avec une bonne qualité possède une faible valeur de la métrique iAWARE. Cette métrique reflète l'interférence inter-flow et intra-flow, l'instabilité du milieu physique et le temps de transmission de données, contrairement aux métriques mETX et ENT que nous décrivons maintenant.

### 2.6.5 Les métriques mETX & ENT

Une majorité des variantes d'ETX permet d'étendre l'application de la métrique ETX (en se basant sur sa définition originale) dans diverses directions telles que l'utilisation de plusieurs

canaux qui peuvent interférer entre eux (interférence inter-flow) ou avec d'autres canaux (interférence intra-flow), et la variation du débit de transmission de la liaison et la taille des paquets.

Les auteurs de [69] se focalisent sur la précision de la fonction d'estimation de perte sur un lien. Ils partent du fait que les liaisons à faible taux de pertes moyen mais avec une forte variabilité sous certaines conditions ont une faible capacité d'estimation de la moyenne statistique [42]. Deux autres statistiques pour l'estimation de nombre de transmissions sur un lien ont donc été proposées par [69].

- La métrique **modified ETX** (**mETX**) (2007) est définie comme suit :

$$mETX = \exp\left(\mu + \frac{1}{2}\sigma^2\right) \quad (.2.6.38)$$

où  $\mu$  est le taux de pertes des paquets moyen estimé sur un lien et  $\sigma^2$  est la variance du taux de perte. Comme ETX, mETX est additif sur les liens d'un chemin.

- La métrique **Effective Number of Transmission** (**ENT**) (2007) est définie comme suit :

$$ENT = \exp(\mu + 2\delta\sigma^2) \quad (.2.6.39)$$

La variable  $\delta$  donne un degré de liberté supplémentaire par rapport à la métrique mETX. La valeur de ENT dépend du nombre de retransmissions ultérieures, ce qui provoque l'abandon d'une tentative de transmission par le protocole de la couche liaison.

Afin de comparer ces deux métriques par rapport à ETX, des observations réelles d'un réseau maillé sans fil ont été réalisées. Elles indiquent que le taux de pertes moyen de paquets de mETX et ENT pourrait atteindre une réduction de 50%, en comparaison avec ETX.

### 2.6.6 La métrique DBETX

La métrique **Distribution Based Expected Transmission Count** (**DBETX**) est définie dans [70] (2008), afin d'améliorer les performances des réseaux multi-canaux en présence de fading, grâce à une vision complète du canal physique et en utilisant une optimisation inter-couches (cross-layer). En utilisant des mesures sur les liens, DBETX permet aux nœuds d'être capable de :

- Estimer la fonction de densité de probabilité **Probability Density Function** (**PDF**) du SINR mesuré ;
- Calculer le taux d'erreur binaire (**Bit Error Rate** (**BER**)), et par conséquent, calculer le taux d'erreurs de paquet (**Packet Error Rate** (**PER**)) attendu ;
- Estimer le nombre moyen de transmissions requises dans un lien donné à partir du SINR ;
- Déterminer le nombre de transmissions nécessaires en considérant le nombre maximum de retransmissions de la couche MAC ;

- Pénaliser les liens avec perte afin de trouver des routes avec des taux de pertes inférieurs, ce qui revient à favoriser les liens avec des faibles probabilités de perte ;
- Représenter les variations du canal sans fil.

La métrique DBETX pour un lien  $l$  est définie comme suit :

$$DBETX(l) = E[ANT(l)] * \frac{1}{1 - P_{out_{MAC}}} \quad (.2.6.40)$$

où  $P_{out_{MAC}}$  est la probabilité que le lien présentera une probabilité de succès  $P_{suc}(x)$  plus faible que  $P_{limit}$  ( $P_{suc}(x) < P_{limit}$ ) avec :

$$P_{limit} = \frac{1}{MaxRetry} \quad (.2.6.41)$$

où  $MaxRetry$  est le nombre maximum de retransmissions permis par la couche MAC (pour 802.11 :  $MaxRetry$  est de 7 en présence de RTS/CTS handshake). La fonction  $ANT(l)$  (Average Number of Transmission) donne le nombre prévu de retransmissions sur un lien, considérant la valeur de  $MaxRetry$ .  $ANT(l)$  est exprimé comme étant :

$$ANT(l) = \begin{cases} \frac{1}{P_{suc}(x)}; P_{suc}(x) > P_{limit} \\ \frac{1}{P_{limit}}; P_{suc}(x) \leq P_{limit} \end{cases} \quad (.2.6.42)$$

Les auteurs [70] montrent que les performances de DBETX augmentent avec la densité du réseau, car le nombre de connexions augmente c'est pourquoi plus de routes deviennent disponibles. Comme la densité augmente, la réduction plus significative dans le nombre moyen de transmissions est atteint. Pour une densité de 15 nœuds par zone de transmission, les résultats montrent une réduction jusqu'à 26% dans ANT. s

Le calcul de DBETX nécessite une information sur le comportement actuel de la liaison sans fil au lieu du comportement moyen. Néanmoins il est impossible que la couche réseau dispose d'une vision complète sur le médium physique, en raison de la différence dans le temps de travail des différentes couches [71].

## 2.6.7 Les métriques BATD & iBATD

La métrique **B**ottleneck **A**ware **R**outing (**BATD**) [72] (2011) a été proposée afin de traiter les interférences intra-flux. Cette métrique permet de déterminer le taux de pertes d'un lien, les différents débits (data rate transmission) de transmissions de données ainsi que les interférences intra-flux dans un chemin. L'idée de base de la métrique BATD se résume comme suit :

- Elle mesure la durée totale des délais de transmission sur les liens ayant le même rang de détection de porteuse, pour chaque canal indépendant au sein d'un chemin ;

- Elle considère le canal ayant les plus grands délais de transmission comme le canal de goulot d'étranglement dans un chemin. Le délai de transmission de ce canal de goulot d'étranglement permet d'évaluer les performances du chemin.

Par conséquent, étant donné un chemin  $p$  avec  $k$  canaux (canal 0,1 ..., $k$ ), BATD est défini comme :

$$BATD(p) = \begin{cases} \max(ETD_1, ETD_2, \dots, ETD_k) \\ ETD_c = \sum_{i=1}^{N_c} ETT_i \quad 0 \leq c \leq k \end{cases} \quad (.2.6.43)$$

où  $ETD_c$  est le délai prévu de transmission pour le canal  $c$  sur le chemin  $p$ , et  $N_c$  est le nombre de liens sur le canal  $c$  avec le chemin  $p$  (les liens au sein de la même gamme de détection de porteuse). Les auteurs de [72] ont prouvé que les performances de BATD surpassent celles des métriques Hop Count, ETX, ETT and WCETT, en matière de capacité.

Les auteurs de BATD proposent une extension de leur métrique appelée **improved Bottleneck Aware Transmission Delay (iBATD)** [73] (2013). L'idée de base de la métrique iBATD est très similaire à celle de la métrique BATD [72], sauf qu'iBATD utilise la métrique **improved Expected Transmission Time (iETT)** [74] au lieu de la métrique ETT. La métrique iETT permet de représenter le délai de transmission de chaque canal individuel. En plus, iETT considère les différents taux de pertes de lien dans un chemin ainsi que l'overhead de la couche MAC lors du calcul du temps de transmission prévu pour un paquet au lieu d'utiliser tout simplement la formule  $\frac{S}{B}$  (voir Section 2.4.2). En effet, les auteurs de [75] expliquent que  $\frac{S}{B}$  ne permet pas de calculer exactement le temps de transmission prévu d'un lien, car il ne considère pas l'overhead nécessaire de la couche MAC pour chaque transmission de paquets. Les performances d'un chemin sont affectées par l'overhead de la couche MAC, en particulier dans le cas où les paquets de données sont de petites tailles. Alors, la métrique iBATD est plus précise pour représenter les performances du goulot d'étranglement. La métrique iBATD est donnée par la formule suivante :

$$iBATD(p) = \begin{cases} \max(ETD_1, ETD_2, \dots, ETD_k) \\ ETD_c = \sum_{i=1}^{N_c} iETT_i \quad 0 \leq c \leq k \end{cases} \quad (.2.6.44)$$

avec iETT tel qu'elle est définie dans [75] :

$$iETT = \sum_{i=1}^n (a_i x + b_i) * (ETX_i) + LID_l \quad (.2.6.45)$$

et où  $a$  et  $b$  sont deux paramètres dépendant des débits de données et des schémas de modulation de la couche MAC. Ils peuvent être facilement dérivées de la méthode décrite dans [76].  $x$  est la taille de la trame en octets, et  $LID_l$  représente approximativement le délai supplémentaire causé

par la différence entre le lien avec le taux de pertes le plus élevé et le lien avec le taux de pertes le plus faible.

$$LID_l = [\max_{0 \leq j \leq n} (P_j) - \min_{0 \leq k \leq n} (P_k)] * (a_j x + b_j) \quad (.2.6.46)$$

où  $\max(P_j)$  et  $\min(P_k)$  désignent respectivement le taux de pertes maximal et le taux de pertes minimal dans tout le chemin au sein d'un canal.

Plusieurs expérimentations ont été menées dans [73], afin de comparer les performances des deux métriques iBATD et BATD. Les résultats montrent qu'iBATD surpasse BATD, en réduisant le temps de latence moyen de paquets (average packet latency) et en améliorant le débit moyen du réseau (average network throughput).

L'étude de [40] décrit une taxonomie des métriques de routage dont elle cite d'autres métriques multi-canaux utilisées dans les réseaux ad hoc. La mobilité des nœuds dans ce type de réseau rend le processus de routage plus complexe à gérer. Plusieurs études dans ce contexte ont été effectuées afin de proposer des métriques sensibles à la mobilité. Nous présentons les métriques de mobilités les plus utilisées dans la section suivante.

## 2.7 Métriques sensibles à la mobilité

Les métriques basées sur la technique de sondage telles que ETX et ses dérivées sont plus efficaces pour les réseaux statiques que la métrique Hop Count. Les métriques de mobilité visent à adapter les routes en temps réel malgré changements fréquents des nœuds. Une large catégorie des métriques utilisent des mesures de la puissance du signal et leurs taux de variation afin de déduire la stabilité des liens et des routes [42]. Les travaux proposés dans le contexte de ABR [34], réutilisés par Route-Lifetime Assessment Based Routing (RABR) [77] ainsi que la métrique Link Affinity [78] sont des exemples de cette catégorie.

### 2.7.1 Les métriques Link Associativity Ticks & Path Aaverage degree of association stability

Les nœuds mobiles transmettent des balises (beacon) de la couche liaison à des intervalles de temps fixe (une seconde comme valeur par défaut). Chaque nœud mesure le nombre reçu des sondes (probe) (associativity ticks) de leurs voisins. Les valeurs mesurées donnent une indication sur la stabilité réelle de la liaison. Si les nœuds mobiles sont dans un état de forte mobilité, alors les valeurs d'associativity tick sont faibles. Par contre, si le nœud mobile est plus stable, alors les valeurs d'associativity tick sont élevées, au-delà d'un certain seuil noté  $A_{thr}$ . Cette métrique prend comme hypothèse le fait que les nœuds alternent entre des périodes de transition/migration et veille (idleness) [42]. Le degré moyen de la stabilité de l'association sur la route  $R$  noté  $A_{ave}^R$ ,

est estimé en fonction des tiques d'associativité (associativity ticks), sur tous les liens le long de la route  $R$  comme suit :

$$A_{ave}^R = \frac{1}{N} \sum_{l \in R} 1_{A_l \geq A_{thr}} \quad (.2.7.47)$$

où 1 est l'indicateur logique et  $n$  est le nombre de liens dans la route  $R$ . Dans [34], la route sélectionnée est tout simplement celle ayant le degré moyen le plus élevé de la stabilité de l'association. La métrique Hop Count est considérée dans le cas où deux routes présentent le même degré moyen de la stabilité de l'association.

### 2.7.2 Les métriques Link Affinity & Path Stability

Link Affinity notée  $l_e$  présente une estimation de la durée de vie d'un lien [78], qui est liée à la variation de la puissance reçue  $P_e$  sur ce lien, en utilisant un seuil  $P_{thr}$  qui détermine si le lien est brisé ou non. La puissance du signal reçu sur  $l_e$  est mesurée pour chaque nœud périodiquement sur chaque intervalle  $dt$ . Soit le taux de variation de la puissance du signal reçue sur  $l_e$  définie comme suit :

$$\Delta P = \frac{(P_e(current) - P_e(previous))}{dt} \quad (.2.7.48)$$

Soit le taux moyen de la variation de la puissance du signal reçue noté  $\Delta P_e^{ave}$ , Link Affinity est définie comme suit :

$$a_e = \begin{cases} high, & \text{if } \Delta P_e^{ave} > 0 \\ \frac{(P_{thr} - P_e)}{\Delta P_e^{ave}}, & \text{if } \Delta P_e^{ave} < 0 \end{cases} \quad (.2.7.49)$$

La stabilité d'une route (Path Stability)  $R$  est définie par :

$$\eta_R = \min_{l \in R} \eta_l \quad (.2.7.50)$$

La stabilité d'une route est donnée à partir du minimum des affinités de tout les liens formant la route. L'affinité entre deux nœuds  $A$  et  $B$  est définie comme suit :

$$\eta_{AB} = \min[a_{AB}, a_{BA}] \quad (.2.7.51)$$

Une route est choisie tant que la valeur estimée pour sa stabilité dépasse le temps nécessaire pour transférer des données L'estimation de cette dernière est égale à la durée nécessaire pour transmettre des données sur la capacité de la liaison notée  $C$ . Un facteur de correction  $f$  est utilisé pour tenir compte de l'imprécision de la métrique, de sorte que le contrôle effectué pour

la route est :

$$D_{AB}/C < f \cdot \eta_{AB} \quad (.2.7.52)$$

La route  $R$  est choisie si l'inégalité est vraie. Sinon la prochaine route disponible est choisie, si elle existe. Dans cette approche, les métriques de liaison sont greffées (PiggyBack) dans les paquets de découverte de route qui se propagent de la source à la destination. Le récepteur prend la décision sur le choix de la sélection de la route à parcourir [42].

### 2.7.3 Les métriques Link Availability & Path Availability

Les auteurs de [79] ont proposé la métrique Link Availability. Cette métrique est définie comme suit :

$$A_{m,n}(t) = Pr(L_{m,n}(t_0 + t) = 1 | L_{m,n}(t_0) = 1) \quad (.2.7.53)$$

où  $L_{m,n}(t) = 1$  indique si la liaison entre le nœud  $m$  et  $n$  est active à l'instant  $t$ . La métrique Link Availability présente la probabilité que le lien est actif à l'instant  $t_0 + t$  étant donné qu'elle est active à l'instant  $t_0$ . Cette métrique ainsi que la métrique Path Availability définie dans [80], permettent de choisir les liaisons les plus stables lorsque les routes sont établies, ce qui permet de réduire les ruptures de liens, et par conséquent de réduire les overheads du réseau. Cette métrique économise aussi la consommation des ressources étant généralement des ressources matérielles dans les réseaux mobiles (portable, téléphone mobile). En effet, Link Availability permet initialement de choisir les bons liens et les bons chemins, pour calculer les routes [81].

Les auteurs de Link Availability ont proposé aussi la métrique Path Availability [80]. Cette métrique définit une mesure probabiliste de la disponibilité des chemins de réseau soumis à des ruptures de liens causées par la mobilité des nœuds dans les réseaux ad hoc. Leur étude est basée sur un modèle de mobilité aléatoire. Chaque nœud est caractérisé par trois valeurs qui décrivent la distribution statistique de la moyenne et la variance de la vitesse d'un nœud, ainsi qu'un intervalle de temps moyen. Avec l'estimation d'un rayon de communication, les auteurs de [80] dérivent une fonction complexe, qui estime la disponibilité prévue d'un lien.

Path Availability dépend de Link Availability,  $A_{m,n}(t)$  entre deux nœuds  $m$  et  $n$ , est définie comme suit :

$$\prod_{m,n}^k(t) = \prod_{i,j \in K} A_{i,j}(t) \quad (.2.7.54)$$

avec  $k$  est le chemin.

Cette métrique permet de connaître sa disponibilité entre deux nœuds  $m$  et  $n$ . Les auteurs de [80] montrent que la métrique Path Availability sélectionne des chemins plus stables et permet

de réduire la charge liée au routage (overhead routing) provoqué par la mobilité des nœuds. Toutefois, cette métrique ne permet pas de savoir exactement la disponibilité d'un chemin du réseau et par conséquent elle ne peut pas induire la réaction du réseau. Les auteurs de [81] confirment que si les probabilités de la disponibilité de la liaison sont bien définies, la métrique Path Availability permettra quand même d'avoir une connaissance importante sur les performances du réseau et la mobilité des nœuds.

#### 2.7.4 La métrique LD

Plusieurs travaux de recherche ont étudié la métrique **Link Duration (LD)** (2003). Une modélisation probabiliste de LD est proposée dans [82], [83], [84]. LD donne la durée de connectivité entre deux nœuds  $m$  et  $n$  à l'instant  $t$  :

$$T_c^m(m, n, t) = e^{-\alpha v(m, n, t)} \quad (.2.7.55)$$

où  $\alpha$  est une constante et  $v(m, n, t)$  est la vitesse relative des nœuds  $m$  et  $n$ . Soit  $n_d$  le nombre de nœuds, la durée moyenne de la liaison  $\overline{T_c^m}$  d'un nœud  $m$  pendant un temps  $T_m$  est :

$$\overline{T_c^m} = \int_0^{T_m} \frac{1}{n_d} \sum_j^{n_d} T_c^m(m, n, t) dt \quad (.2.7.56)$$

Les auteurs de [83] confirment que cette métrique est considérée comme une métrique de mobilité puisqu'elle donne une information sur la durée de la liaison ou le temps de la connexion d'un lien  $T_c^m$ . Ce paramètre indiquant la disponibilité d'un nœud peut donc être utile pour le choix des routes au niveau du protocole de routage, mais aussi au niveau du choix du protocole de routage en lui même. Cependant, cette métrique ne reflète pas la fréquence des changements de l'état de la liaison (transmission possible : état *ON*, transmission non possible : état *OFF*). En particulier, si la durée de l'état *OFF* est inférieure à la durée d'expiration des routes,  $T_c^m$  reste inchangé.

Une définition plus simple de la métrique LD est présentée dans [85], [6], dont la formule est la suivante :

$$LD(m, n, t_1) = (t_2 - t_1) \quad (.2.7.57)$$

Cette métrique est calculée en mesurant la durée de vie d'une liaison entre deux nœuds  $m$  et  $n$  à l'instant  $t_1$ . Pour chaque lien existant, elle donne le temps à partir de la détection de la liaison. Tant que les deux nœuds  $m$  et  $n$  sont à portée radio l'un de l'autre, alors LD augmente.

La figure 2.5 illustre un exemple de calcul de la métrique LD pour le nœud C. Initialement, à  $T = 10$  secondes, les nœuds mobiles A et B sont à portée radio de C, par contre les nœuds mobiles



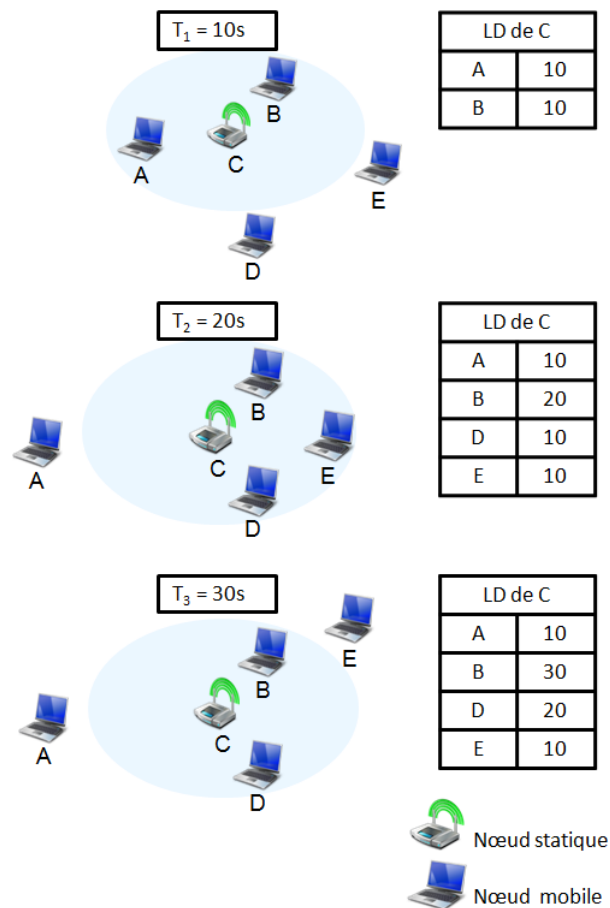


FIGURE 2.5 – Exemple de calcul de la métrique LD

E et D sont hors portée radio de C. Alors, LD pour A et B est de 10 secondes. Ensuite, à  $T = 20 - 10$  secondes, la topologie change. Les nœuds mobiles A, E et D bougent : A sort de la portée radio de C alors la valeur de LD pour A ne change pas, tandis que E et D entrent dans la portée radio de C, la valeur de LD pour E et D est de 10 secondes. Le nœud B est toujours dans la portée radio de C, LD pour B est incrémenté de 10 et devient 20 secondes. Après, à  $T = 30 - 20$  secondes, le nœud E bouge et sort de la portée radio de C. Le nœud A reste hors portée radio du nœud C, LD pour A et E reste inchangée. Par contre, LD pour B et D est incrémentée de 10 secondes.

Les auteurs de [86] intègrent la métrique LD dans le protocole de routage LAR et valide son intérêt par simulation avec le modèle de mobilité Random Way Point [87]. Ils montrent que lorsque la valeur de LD augmente, le taux de livraison de paquet (PDR : Packet Delivery Ratio) augmente aussi, et lorsque la valeur de LD diminue, alors le taux de livraison de paquet diminue. Dans le même contexte, [85] montrent que LD peut être un outil utile pour gérer la mobilité. Ils considèrent la durée d'un chemin (Path Duration) comme un critère pour sélectionner les routes. La durée d'un chemin est tout simplement le minimum des durées (LD) des différents liens formant ce chemin. Plus la durée du chemin est longue, plus le chemin est stable conduisant

à minimiser les ruptures des liens en cas de mobilité [88]. Tandis que les auteurs de [84] ainsi que [83] révèlent que LD ne donne pas une vision sur l'impact de la fréquence des changements des liens.

### 2.7.5 La métrique Link Change Rate, Link State Changes $\lambda_{lc}$

Cette métrique est appelée Link Change Rate dans [86] (2002) et Link State Changes (2004) dans [83].  $\lambda_{lc}$  définit la fréquence des changements d'état d'une liaison. En considérant le calcul de  $\lambda_{lc}$ , il est possible de capturer les changements des liens dans le réseau avec le modèle de mobilité Random Way Point [87].  $\lambda_{lc}$  peut être modélisé comme l'inverse de la moyenne de la période de la connexion et de la rupture d'un lien [89].

$$\overline{\lambda_{lc}} = \frac{1}{\overline{T_{lc}}} \quad (.2.7.58)$$

Dans la littérature, cette métrique a été proposée dans l'objectif de pouvoir effectuer une comparaison entre les résultats de diverses études sur différents modèles de mobilité. Cette métrique a été utilisée aussi pour évaluer l'éloignement comme une nouvelle mesure de mobilité [90]. Pour étudier les performances des différents protocoles MANET, les auteurs de [86] visent à créer un environnement normalisé en utilisant la métrique Link change rate. Elle permet d'observer la fréquence des changements d'état de la liaison, aspect non traité par la métrique LD. Mais, elle ne permet pas d'avoir une vision globale de la disponibilité d'un nœud, et donc ne permet pas d'évaluer l'effet des changements d'état de liaison sur la transmission [83].

### 2.7.6 La métrique Link Stability $L_s$

En combinant les avantages de la métrique Link Duration et la métrique Link State Changes, [83] propose la métrique  $L_s$  (2004). Avec le modèle de mobilité Random Way Point, les deux paramètres  $\lambda_{lc}$  et  $T_c^i$  ne peuvent pas décrire correctement la mobilité [79]. En effet, l'augmentation de la vitesse maximale augmente  $\lambda_{lc}$  mais ne perturbe pas  $T_c^i$ . En plus, lorsque la portée d'un nœud augmente,  $T_c^i$  augmente, par contre  $\lambda_{lc}$  reste insensible à cet effet. Il est donc nécessaire d'introduire une métrique qui soit à la fois sensible à la durée de la liaison et à la fréquence des changements de l'état de la liaison. Ces deux paramètres traitent deux problématiques différentes mais elles sont complémentaires et rassembler leurs avantages dans une même métrique permet de combler leurs faiblesses [81].  $L_s$  est donc introduite en tenant compte de ses caractéristiques liées à la mobilité du réseau.

$$l_s = \frac{\overline{T_c}}{\overline{\lambda_{lc}}} \quad (.2.7.59)$$

Pour l'évaluation des performances de  $l_s$  avec le modèle de mobilité Random Way Point, [89] montre qu'elle est meilleure que les autres. Leurs mesures de performances sont basées sur la capacité de la représentation de la mobilité, plutôt que sur les performances du protocole. Les auteurs de [89] montrent que  $l_s$  reflète bien le modèle de mobilité de façon aléatoire. Par contre les auteurs de [6], montrent le contraire. À partir de plusieurs simulations avec différents protocoles et différents modèles de mobilité, ils montrent que la métrique  $LD$  est meilleure que les métriques  $l_s$  et  $l_c$ .

## 2.8 Taxonomie des métriques de routage

Les travaux de [39], [40], [41], [42] spécifiques à la classification des métriques de routage, nous ont permis d'enrichir notre classification. Dans le tableau 2.10, une taxonomie des métriques de routage sur la base des critères suivants est illustrée :

- A - Catégorie de la métrique : spécifique à la topologie, basée sur la mesure de la puissance du signal, qui reflètent la qualité de la liaison, d'équilibrage de la charge de trafic, multi-canaux et sensibles à la mobilité ;
- B - Objectifs d'optimisation de la métrique : minimiser le délai, maximiser la durée de vie de route, maximiser la probabilité de livraison et équilibrer la charge de trafic par une distribution équitable ;
- C - Méthode utilisée pour obtenir l'information nécessaire pour calculer la métrique d'un lien : utilisation des informations locales, contrôle passif, sondage actif et sondage Piggy-Backed ;
- D - Fonction utilisée pour calculer la métrique de routage tout le long d'un chemin : somme, multiplication ou mesures statistiques ;
- E - Propriétés relatives pour chaque métrique : elle reflète le taux de pertes de paquet, le débit de transmission, l'interférence inter-flux, l'interférence intra-flux, ou elle est isotonique ;
- F - Plateforme pour laquelle la métrique a été conçue : réseaux ad hoc sans fil multi-sauts, réseaux ad hoc sans fil multi-radios, Réseaux ad hoc sans fil à débit multiple, réseaux ad hoc sans fil variant dans le temps, canaux sans fil avec fading et réseaux ad hoc sans fil mobile (MANET).

## 2.9 Choix des métriques

Le tableau 2.10 présente une synthèse des caractéristiques des métriques bibliographiques étudiées. Nous utilisons ce tableau pour préciser les métriques qui peuvent être utilisées pour la

conception d'une nouvelle métrique qui reflète la qualité du lien et qui gèrent la mobilité, en même temps. Les métriques à étudier doivent vérifier les conditions suivantes :

1. Calculable dans un environnement distribué sans connaissances globales du réseau ;
2. Calculable dans une implémentation réelle d'un réseau (testbed), c'est à dire que les paramètres doivent pouvoir être mesurés ou disponibles au niveau du driver ;
3. Indépendante du protocole de routage utilisé et indicatrice des performances du protocole.

Le but de cette thèse est de proposer des nouvelles métriques qui soient à la fois des métriques de qualité du lien et sensibles à la mobilité. L'étude bibliographique effectuée dans ce chapitre, montre qu'une grande partie des métriques étudiées sont des dérivées des deux métriques de base ETX et ETT. Ces métriques sont des métriques de qualité de lien. En effet, ETX et ETT donnent des valeurs pertinentes sur la qualité de lien [53], précisément le taux de pertes et la capacité d'un lien. Une étude comparative entre Hop Count, ETX et ETT dans le cadre d'un réseau mobile, serait intéressante pour voir l'impact de la mobilité sur leurs performances.

Nous nous intéressons aussi aux métriques sensibles à la mobilité. L'étude de [6] sur l'adaptation à la mobilité dans les réseaux ad hoc, analyse les performances des métriques de mobilité suivantes : la durée de lien (LD), la fréquence de changement de lien ( $\lambda_{lc}$ ) et la stabilité de liens ( $L_s$ ). Les auteurs préconisent l'utilisation de la métrique LD, en raison de sa simplicité de calcul, de sa relative indépendance vis à vis du modèle de mobilité et de sa forte influence sur les performances des protocoles de routage.

En conclusion, à l'égard de l'objectif de conception de la nouvelle métrique, nous avons étudié principalement dans cette thèse les 4 métriques suivantes : Hop Count, ETX, ETT, et LD.

## 2.10 Conclusion

Nous avons défini la notion de métrique de routage dans les réseaux ad hoc, précisé ses objectifs d'optimisation et présenté sa méthode de calcul. Nous définissons six catégories de métriques : de topologie, basée sur la mesure de la puissance du signal, qui reflètent la qualité de la liaison, d'équilibrage de la charge de trafic, multi-canaux, et sensibles à la mobilité. Dans l'objectif de retenir des métriques qui soient les plus universelles possibles, nous retenons les métriques qui reflètent la qualité de lien et qui prennent en compte la mobilité. Plus précisément nous retenons quatre métriques : Hop Count, ETX, ETT et LD. Nous poursuivons l'étude de ces métriques par une analyse quantitative dans le chapitre suivant et une comparaison expérimentale.

TABLE 2.5 – Taxonomie des métriques : Objectifs d’optimisation

Métriques	Objectifs d’optimisation			
	Minimise le délai	Maximise la durée de vie de route	Maximise la probabilité de livraison	Distribution équitable de la charge de trafic
<b>Topologie</b> Hop Count (2003) [3]	×			
<b>Puissance du signal</b> Facteur de qualité de lien (1999) [54] Routage préemptif (2001) [55] SSAR (1997) [56]		× × ×		
<b>Qualité de lien</b> ETX (2003) [4] ETT (2004) [5] MTM (2004) [58]	× × ×		× × ×	
<b>Charge de trafic</b> EAB (2010) [59] MF-Transmission (2011) [60] C2WB (2011) [61] MIND (2009) [62]	× × × ×		× × × ×	× × × ×
<b>Multi-canaux</b> ModWCCETT (2014) [64] MIC (2005) [66] MCR (2006) [67] iAWARE (2006) [68] mETX (2007) [69] ENT (2007) [69] & DBETX (2008) [70] iBATD (2013) [73]	× × × × × × ×		× × × × × × ×	×



TABLE 2.7 – Taxonomie des métriques : Propriétés

Métriques	Propriétés				
	Taux de pertes de paquet	Débit de transmission	Interférence inter-flux	Interférence intra-flux	Isotonique
<b>Topologie</b> Hop Count (2003) [3]					
<b>Puissance du signal</b> Facteur de qualité de lien (1999) [54] Routage préemptif (2001) [55] SSAR (1997) [56]					
<b>Qualité de lien</b> ETX (2003) [4] ETT (2004) [5] MTM (2004) [58]	× × ×	 × ×			× × ×
<b>Charge de trafic</b> EAB (2010) [59] MF-Transmission (2011) [60] C2WB (2011) [61] MIND (2009) [62]	× × × ×	×  × ×	× × ×	×  ×	× × ×
<b>Multi-canaux</b> ModWCCETT (2014) [64] MIC (2005) [66] MCR (2006) [67] iAWARE (2006) [68] mETX & ENT (2007) [69] DBETX (2008) [70] iBATD (2013) [73]	× × × × × × ×	× × × ×  ×	×   ×	× × × ×  ×	    × × ×

TABLE 2.8 – Taxonomie des métriques : Propriétés (suite)

Métriques	Propriétés			
	Taux de pertes de paquet	Débit de transmission	Interférence inter-flux	Interférence intra-flux
<div>Mobilité</div> <div>ABR (1997) [34]</div> <div>Link Associativity ticks [42]</div> <div>Link Affinity &amp; Path Stability (1999) [78]</div> <div>Link Availability &amp; Path Availability (2006) [79]</div> <div>Link duration (LD) (2003) [85]</div> <div>Link change rate (2002) [86] &amp; Link state changes [83]</div> <div>Link Stability (2004) [83]</div>				



TABLE 2.9 – Taxonomie des métriques : faisabilité

Métriques	Méthode de calcul				Métrique du chemin			Plateforme					
	Utilisation des informations locales	Contrôle passif	Sondage actif	Sondage PiggyBacked	Somme	Multiplication	Mesures statistiques	Réseaux ad hoc sans fil multi-sauts	Réseaux ad hoc sans fil multi-radios	Réseaux ad hoc sans fil à débit multiple	Réseaux ad hoc sans fil variant dans le temps	Canaux sans fil avec fading	Réseaux ad hoc sans fil mobile (MANET)
<b>Topologie</b> Hop Count (2003) [3]	×				×			×					
<b>Puissance du signal</b> Facteur de qualité de lien (1999) [54] Routage préemptif (2001) [55] & SSAR (1997) [56]	×							×	×				
<b>Qualité de lien</b> ETX (2003) [4] ETT (2004) [5] & MTM (2004) [58]			×	×	×			×	×	×			
<b>Charge de trafic</b> EAB (2010) [59] MF-Transmission (2011) [60] C2WB (2011) [61] MIND (2009) [62]	×	×	×		×			×	×	×			

TABLE 2.10 – Taxonomie des métriques : faisabilité (suite)

Métriques	Méthode de calcul				Métrique du chemin			Plateforme					
	Utilisation des informations locales	Contrôle passif	Sondage actif	Sondage PiggyBacked	Somme	Multiplication	Mesures statistiques	Réseaux ad hoc sans fil multi-sauts	Réseaux ad hoc sans fil multi-radios	Réseaux ad hoc sans fil à débit multiple	Réseaux ad hoc sans fil variant dans le temps	Canaux sans fil avec fading	Réseaux ad hoc sans fil mobile (MANET)
<b>Multi-canaux</b>  ModWCCEIT (2014) [64]  MIC (2005) [66] & MCR (2006) [67]  iAWARE (2006) [68] mETX & ENT (2007) [69] DBETX (2008) [70] iBATD (2013) [73]	×		×		×			×	×	×	×	×	
			×		×			×	×	×	×	×	
<b>Mobilité</b>  ABR (1997) [34]  Link Associativity ticks [42] Link Affinity & Path Stability (1999) [78] Link Availability & Path Availability (2006) [79] Link duration (LD) (2003) [85] & Link Stability (2004) [83] Link change rate (2002) [86] & Link state changes [83]	×		×	×	×								×
			×	×	×								×

## Chapitre 3

# Implémentation & Évaluation des métriques ETX, ETT et LD

### Sommaire

---

<b>3.1</b>	<b>Calcul &amp; implémentation des métriques ETX &amp; ETT</b>	<b>67</b>
3.1.1	ETX	67
3.1.1.0.1	Le processus des messages HELLO et TC	68
3.1.2	ETT	71
3.1.2.1	Algorithmes existants	71
3.1.2.2	Approche proposée	72
3.1.2.2.1	Minstrel	72
<b>3.2</b>	<b>Calcul &amp; implémentation de la métrique LD</b>	<b>73</b>
<b>3.3</b>	<b>Topologies de référence</b>	<b>75</b>
3.3.1	Topologie chaîne	75
3.3.2	Topologie mesh	76
3.3.3	Topologie ad hoc	76
<b>3.4</b>	<b>Évaluation des performances</b>	<b>77</b>
3.4.1	Configuration de la simulation	77
3.4.2	Résultats pour la chaîne de noeud	78
3.4.3	Résultats pour le réseau mesh	81
3.4.4	Résultats pour la topologie aléatoire	84
<b>3.5</b>	<b>Conclusion</b>	<b>87</b>

---

Dans le chapitre précédent, nous avons justifié notre choix pour l'étude des quatre métriques suivantes : Hop Count, ETX, ETT, et LD. Le protocole OLSR natif utilise par défaut la métrique Hop Count dans le processus de calcul de route. Nous avons implémenté les trois dernières métriques : ETX, ETT et LD pour pouvoir les comparer. Dans ce chapitre, nous présentons notre

propre implémentation de ces métriques dans le protocole OLSR. Ensuite, nous décrivons le processus de mise à jour des métriques dans le réseau, dans le cas d'un changement de topologie : mobilité, rupture de lien, etc. Puis, nous détaillons la méthode utilisée pour propager les métriques dans le réseau. Finalement, nous étudions le comportement des différentes métriques en présence de mobilité. Des simulations approfondies ont été effectuées pour comparer ces métriques.

### 3.1 Calcul & implémentation des métriques ETX & ETT

#### 3.1.1 ETX

La métrique ETX a été proposée dans le but de palier les inconvénients de la métrique Hop Count qui ne prend pas en considération des critères de la qualité du lien tel que le taux de perte, le débit de transmission, la puissance du signal reçue, etc. ETX prend en compte le taux de perte d'un lien comme critère de qualité de lien car elle estime le nombre de transmission et de retransmission prévues pour la bonne réception d'un paquet sur un lien. Nous rappelons l'équation d'ETX entre deux nœuds A et B (comme mentionné dans la partie 2.4.1) :

$$ETX_{AB} = \frac{1}{df * dr} \quad (.3.1.1)$$

où  $df$  est le taux de perte de A vers B, et  $dr$  est l'inverse du taux de perte de B vers A. Dans notre implémentation,  $df$  et  $dr$  présentent respectivement le PDR de A vers B, et le PDR de B vers A. Ces quantités peuvent être estimées facilement sur la base des échanges périodiques des messages HELLO entre les nœuds voisins sur un intervalle de temps.  $df$  et  $dr$  peuvent être estimé sur une fenêtre  $[T, T+t]$  comme suit :

$$df[T, T+t] = PDR_{AB[T, T+t]} = \frac{\text{Nombre total de message HELLO}_{[T, T+t]} \text{ recus par B}}{\text{Nombre total de message HELLO}_{[T, T+t]} \text{ envoyes par A vers B}} \quad (.3.1.2)$$

$$dr[T, T+t] = PDR_{BA[T, T+t]} = \frac{\text{Nombre total de message HELLO}_{[T, T+t]} \text{ recus par A}}{\text{Nombre total de message HELLO}_{[T, T+t]} \text{ envoyes par B vers A}} \quad (.3.1.3)$$

Selon la définition d'ETX, si  $df * dr$  augmente le PDR du lien en question augmente aussi. En d'autres termes, plus la valeur d'ETX est petite, plus la qualité du lien en terme de taux de perte est meilleure. La valeur idéale de la métrique ETX est 1, qui implique que le lien en question est sans perte et que le PDR dans les deux sens est à 100%. Une route utilisant ETX comme métrique est définie comme la somme de tout les ETX correspondant aux liens formant cette route. La route optimale est la route qui minimise la somme des ETX.

Comme les métriques ETX ou ETT sont additives, nous avons implémenté l'algorithme Dijkstra [18] dans OLSR, celui ci utilisait l'algorithme de Bellman Ford. L'algorithme Dijkstra trouve

le chemin qui minimise la somme des métriques (sur un chemin). En outre, nous intégrons le calcul des métriques au démon OLSR.

L'utilisation des messages HELLO pour estimer le taux de perte d'un lien est plus simple et plus facile que l'utilisation des paquets de données. En effet, le démon de routage n'a pas de statistiques sur l'utilisation du lien.

Par contre, A peut estimer la qualité du lien (A-B) à partir des messages HELLO échangés entre eux. Nous expliquons dans le paragraphe suivant la méthode utilisée pour calculer la métrique ETX grâce aux messages HELLO.

### 3.1.1.0.1 Le processus des messages HELLO et TC

L'implémentation est réalisée avec le simulateur *NS-3* [91]. Les messages HELLO permettent d'estimer le taux de perte d'un lien. Nous avons utilisé un tableau multidimensionnel noté HELLO pour enregistrer les messages HELLO reçus entre chaque nœud et son voisin. Ce tableau prend comme premier indice l'adresse du nœud qui a reçu le HELLO, comme deuxième indice l'adresse du nœud qui a envoyé le HELLO, et comme dernier indice le temps de réception du message HELLO. Ce tableau s'incrémente à chaque réception d'un message HELLO d'une source à une destination à l'instant T.

---

#### Algorithm 1: Management Mobility ALGORITHM

---

```

void Mobility_Management()
begin
    /* The different variables are updated at each HELLO reception */
    Update CURRENT_TIME of this HELLO;
    Update STATE of neighbors;
    LIMIT_TIME[R][S] = CURRENT_TIME + 1.5 * HELLO_INTERVAL;

    /* For each neighbor in the NEIGHBORSET of R */
    for N ∈ NEIGHBOR_SET do
        /* When the state of the link between R and his neighbor N is symetric */
        if (CURRENT_TIME > LIMIT_TIME[R][N] AND STATE[R][N] == SYM) then
            /* HELLO message was lost */
            HELLO[R][N][TIME[R][N]] = 0;
            COMPUTE_ETX(R,N);
            TIME[R][S] = (TIME[R][S] + 1) % HELLO_NUMBER;
            LIMIT_TIME[R][S] = CURRENT_TIME + 1.5 * HELLO_INTERVAL;

        /* HELLO message was received */
        TIME[R][S] = (TIME[R][S] + 1) % HELLO_NUMBER;
        HELLO[R][N][TIME[R][N]] = 1;
        COMPUTE_ETX(R,S);

```

---

L'algorithme 1 présente l'algorithme de mise à jour de la matrice HELLO pour chaque réception d'un message HELLO entre deux nœuds R et S. Nous définissons un temps limite pour la bonne réception d'un message HELLO noté *LIMIT\_TIME*. Lorsque *CURRENT\_TIME* relative à un

voisin N symétrique<sup>1</sup> du nœud R a dépassé *LIMIT\_TIME*, le message HELLO entre R et N est considéré comme perdu. Dans ce premier cas, nous faisons une mise à jour du tableau HELLO relative à R et N à l'instant  $TIME[R]/[N]$  avec la valeur 0. Ensuite, nous calculons la métrique ETX pour ce lien avec la fonction  $COMPUTE\_ETX(R,N)$  (voir l'algorithme 2) que nous allons détailler dans le paragraphe suivant. Dans le deuxième cas, lorsque un HELLO est bien reçu entre R et S, le tableau HELLO à l'instant  $TIME[R]/[S]$  est mis à jour avec la valeur 1. Cette mise à jour sera prise en compte dans le calcul de la métrique ETX entre R et S par la fonction  $COMPUTE\_ETX(R,S)$ .

L'algorithme 2 résume le calcul de la métrique ETX entre deux nœuds R et S. Comme ETX est une métrique symétrique, la valeur de cette métrique entre deux nœuds R et S est la même dans les deux sens. Nous utilisons une fenêtre glissante dans le calcul de la métrique ETX, pour prendre en compte les valeurs les plus pertinentes. Cette fenêtre nous permet d'avoir la moyenne de la métrique ETX par rapport à un nombre fixe de message HELLO noté *HELLO\_NUMBER*. Nous discutons le choix du paramètre *HELLO\_NUMBER* dans le paragraphe suivant.

---

**Algorithm 2:** Compute ETX ALGORITHM
 

---

```

void Compute_ETX()
begin
    /* Initialization of the different variables */
    MEAN_HELLO_R = 0;
    MEAN_HELLO_S = 0;
    DF_RS = 0;
    DR_SR = 0;
    for i ← 0 to HELLO_NUMBER do
        MEAN_HELLO_R+ = HELLO[R][S][i];
        MEAN_HELLO_S+ = HELLO[S][R][i];
    DF_RS = MEAN_HELLO_R/HELLO_NUMBER;
    DR_SR = MEAN_HELLO_S/HELLO_NUMBER;
    ETX[R][S] = 1/(DF_RS*DR_SR);
    /* ETX[S][R] = 1/(DF_SR*DR_RS) */
  
```

---

Le paramètre *HELLO\_NUMBER* désigne le nombre de messages HELLO pour lequel nous calculons la moyenne d'ETX pour un lien donné. Nous avons testé deux valeurs pour ce paramètre : *HELLO\_NUMBER* = 3 et *HELLO\_NUMBER* = 10. Nous comparons ces deux valeurs au travers de simulation dans la Section 3.4.

Afin de propager la métrique ETX au voisinage de chaque nœud, nous avons pensé à étendre le format des paquets HELLO du protocole OLSR pour ajouter un champ spécifique à la valeur de la métrique. Les auteurs de [93] ont déjà appliqué la même procédure afin d'implémenter la métrique AIRTIME [94] dans OLSR.

---

1. Un nœud X est dit voisin symétrique d'un nœud Y s'il existe une fréquence radio f sur laquelle X entend Y et Y entend X, le lien (X,Y) est dit lien symétrique [92]

Nous avons ajouté le champ “Neighbor Interface Metric” spécifique à la valeur de la métrique pour chaque “neighbor interface main Address”. La valeur de la métrique est ajoutée pour chaque voisin du nœud en question. De cette manière chaque nœud de la topologie envoie la métrique correspondante avec son voisin dans le message HELLO.

L’extension du message HELLO avec la valeur de la métrique correspondante sert tout simplement à propager cette information dans le voisinage de chaque nœud mais cette technique ne sert pas à propager cette information dans tout le réseau. Pour se faire, nous avons fait étendre les messages TC puisque ces derniers sont responsables sur la propagation des informations topologiques à travers tout le réseau. Le champ “Advertised Neighbor Main Address” contient l’adresse principale d’un nœud au nœud expéditeur, comme définie dans RFC 3626 [28]. Toutes les adresses principales des voisins du nœud expéditeur sont mises dans le message TC. Nous avons ajouté le champ métrique noté “Advertised Neighbor Metric” pour chaque adresse principale du voisin du nœud expéditeur. Ces extensions dans les messages TC et HELLO ont été considérées dans la version optimisée OLSRv2 [95]. Cette version définie dans RFC 7181 est le successeur d’OLSR (RFC 3626). Par rapport à OLSR, OLSRv2 conserve les mêmes mécanismes et algorithmes de base, mais il offre la possibilité d’utiliser une métrique de lien dans le choix des routes les plus courtes.

Le protocole OLSR définit des bases de données appelées “Tuples” essentielles pour la gestion des liens, des voisins et pour le contrôle des informations topologiques comme c’est détaillé dans la Section 1.4. Il était nécessaire de changer aussi ces bases de données dans le but de prendre en compte la valeur de la métrique, de la manière suivante :

- Link\_Tuple (L\_local\_iface\_addr, L\_neighbor\_iface\_addr, L\_SYM\_time, L\_ASYM\_time, L\_time, **L\_metric**)
- Neighbor\_Tuple (N\_neighbor\_main\_addr, N\_status, N\_willingness, **N\_metric**)
- Topology\_Tuple((T\_dest\_addr, T\_last\_addr, T\_seq, T\_time, **T\_metric**)

Pour la première base de données Link\_Tuple, nous avons ajouté la valeur de la métrique, pour chaque lien symétrique puisque ETX (ou ETT) est une métrique symétrique. Aussi cette valeur est ajoutée pour les voisins dans la base de données Neighbor\_Tuple afin de la propager entre les voisins d’un nœud. La dernière base de données modifiée est Topology\_Tuple qui est primordiale pour la propagation de la métrique dans toute la topologie.

Une fois que le lien entre deux nœuds devient asymétrique ou non spécifié ou perdu, il sera automatiquement supprimé de la base de données utilisée dans le processus de calcul de route. De cette manière, le lien en question ne sera pas pris en considération sauf si le lien se crée encore une fois et devient symétrique.

### 3.1.2 ETT

Deux approches ont été proposées pour l'implémentation de la métrique ETT. Dans le paragraphe suivant, nous détaillons ces deux approches. Ensuite, nous présentons notre propre approche.

#### 3.1.2.1 Algorithmes existants

La première approche a été définie par [64] dans le but d'estimer la valeur de  $B$ . Ils mesurent la largeur de la bande passante en utilisant la technique de paquets pairs [96]. Cette technique utilise deux paquets sonde (back-to-back), de tailles différentes. Le premier paquet de sonde est avec une faible taille de 137octets, tandis que le deuxième paquet de sonde est avec une taille plus grande de 1137octets.

Ces deux paquets de sonde sont envoyés par chaque nœud à chacun de leurs voisins, à chaque minute. Chaque voisin mesure le temps d'inter-arrivée entre les deux sondes en mesurant le délai entre la réception du premier et du deuxième paquet, et ensuite, il communique la valeur de retour à l'expéditeur. Après la réception d'un nombre prédéterminé d'échantillons de délai (le minimum étant 10 échantillons consécutifs), l'émetteur estime la largeur de la bande passante en divisant la taille de la deuxième sonde par le plus petit délai obtenu d'échantillon. Les auteurs de [64] expliquent que cette estimation néglige plusieurs facteurs qui influencent le délai de livraison de paquets. Elle est donc considérée comme une estimation pas très précise. Mais, elle peut malgré tout servir à distinguer les différentes bandes passantes des liens.

La deuxième approche proposée est définie par [97]. Les auteurs utilisent les paquets de données et les trames ACK du IEEE 802.11 afin d'estimer le taux de perte. L'idée de base est de calculer périodiquement le taux de perte des paquets de données et des trames ACK pour chaque voisin. Pour estimer le taux de perte des données, ils diffusent un certain nombre de paquets de la même taille que les trames de données, un paquet pour chaque débit de données tel que défini dans la norme IEEE 802.11. Pour estimer le taux de perte des trames ACK, ils diffusent de petits paquets de la même taille que les trames ACK transmis à une vitesse de base (débit utilisé pour les ACK). Selon [97], la métrique ETT considère le meilleur débit à atteindre noté  $r_i$  et la probabilité de distribution de paquets ACK dans le sens inverse noté  $p_{ACK}$ . Ainsi, ETT peut s'écrire sous cette forme :

$$ETT_i = 1 / (r_i * p_{ACK}) \quad (.3.1.4)$$



Ces deux approches ont été proposées dans le cadre d'une implémentation d'ETT dans un test-bed.

### 3.1.2.2 Approche proposée

Dans notre étude, nous utilisons le simulateur *NS-3* pour valider notre approche par simulation. Alors pour implémenter ETT dans OLSR avec *NS-3*, nous avons choisi d'utiliser les informations utiles de la couche physique qui peuvent nous donner une idée sur la capacité d'une liaison. Nous avons utilisé les informations disponibles à partir de l'algorithme de contrôle de débit (*WIFI\_MANAGER*) qui fixe le débit physique auquel les trames de données sont envoyés. *NS-3* utilise plusieurs algorithmes de contrôle de débit dont nous pouvons citer : AARF, AARFCD, AMRR, ARF, Constant-rate, Minstrel, etc.

Le *WIFI\_MANAGER* utilisé dans notre étude est Minstrel. Nous le présentons donc plus en détail au paragraphe suivant.

#### 3.1.2.2.1 Minstrel

Minstrel est un algorithme de contrôle de débit 802.11, porté entre autres sur le pilote MAD wifi [98]. Cet algorithme prend en charge plusieurs tentatives de débits. Minstrel adopte une approche heuristique basée sur l'expérimentation, l'évaluation ou les méthodes d'essais et d'erreurs. Il a recours dans sa mise en œuvre à une table de débit pour chacun des nœuds distants étant en communication.

Minstrel définit la mesure de la probabilité de réussite de transmission de paquets comme débit, de la manière suivante :

$$throughput = \frac{Prob\_success\_transmission * Mega\_bits\_transmitted}{time\ for\ 1\ try\ of\ 1\ packet\ to\ be\ sent\ on\ the\ air} \quad (.3.1.5)$$

Où *Prob\_success\_transmission* est la probabilité de réussite de transmission de paquet, et *Mega\_bits\_transmitted* est le nombre de bits transmis.

Cette mesure de probabilité de réussite règle la vitesse de transmission afin d'obtenir le nombre maximal de bits de données à travers l'interface radio. Le module enregistre le succès de tous les paquets transmis. À partir de ces données, le module décide de l'efficacité d'un débit physique par rapport aux autres. Cependant, Minstrel utilise aussi un élément de variabilité, afin de forcer le module à examiner les débits de bits autres qu'optimale. Par conséquent, certains pourcentages des paquets doivent être envoyés à des taux considérés comme non optimale.

Minstrel utilise une minuterie qui se déclenche (10 fois par seconde) pour mettre à jour le tableau des statistiques. Le noyau de cet algorithme de contrôle de débit est EWMA (Exponential Weighted Moving Average). EWMA permet de donner plus d'importance aux résultats récents, qu'aux anciens. Par conséquent, les changements environnementaux sont pris en considération, et les anciennes données sont ignorées. EWMA est utilisée pour le traitement de l'historique de la réussite de chaque débit. À l'issue du calcul, une décision est prise quant à la vitesse qui a le meilleur débit, le deuxième meilleur débit et la plus forte probabilité de succès. Ces données sont utiles afin de remplir la chaîne de tentatives (pour les 100 prochaines ms).

L'algorithme Minstrel collecte des statistiques de toutes les tentatives de paquets. Il consacre un certain pourcentage de paquet pour faire un "look around" ("regarder autour") : il essaye au hasard d'autres débits, et il recueille des statistiques. 10% est le pourcentage par défaut des trames consacrées pour faire le "look around" [98].

Une fois que nous extrayons l'information sur le débit échangé du Minstrel (couche physique) et la taille des paquets des données, la procédure de calcul de la métrique ETT est la même que pour ETX. Nous changeons juste la formule de calcul et la suite du traitement est la même.

Nous avons décrit dans cette section les étapes de la mise en oeuvre des métriques ETX et ETT, nous détaillons dans la section suivante la méthode de calcul et de l'implémentation de la métrique LD.

### 3.2 Calcul & implémentation de la métrique LD

La métrique LD est définie comme une métrique de mobilité. LD est fixé en fonction de la durée de vie du lien : la différence entre le temps actuel et la dernière fois que le lien a été défini comme symétrique.

Chaque couple formé d'un nœud expéditeur noté  $S$  et son voisin direct noté  $R$  est appelé Tuple. Si la liaison de ce Tuple est symétrique, la durée de la liaison en seconde est enregistrée dans une variable notée *LIFE\_TIME*. Ensuite, la valeur de la métrique LD de ce tuple noté  $LD\_METRIC[R][S]$  est affectée à *LIFE\_TIME*. Enfin, une mise à jour de ce Tuple sera exécutée pour prendre en compte les nouvelles informations. La métrique LD est propagée dans le réseau de la même manière que les deux métriques ETX et ETT grâce aux messages HELLO et TC.

La figure 3.1 illustre un exemple de calcul de la métrique LD pour une topologie donnée. Nous supposons que la communication entre les nœuds a commencé à un temps  $T = 0s$  et les liens symétriques tels que (A, B) et (B, E) se sont créés.

À  $T = 10s$ , nous remarquons que les liens symétriques existent encore et leurs durées de vie 10s, c'est pour cette raison que  $LD(A, B) = 10$  et  $LD(B, E) = 10$  (voir figure 3.1(a)).

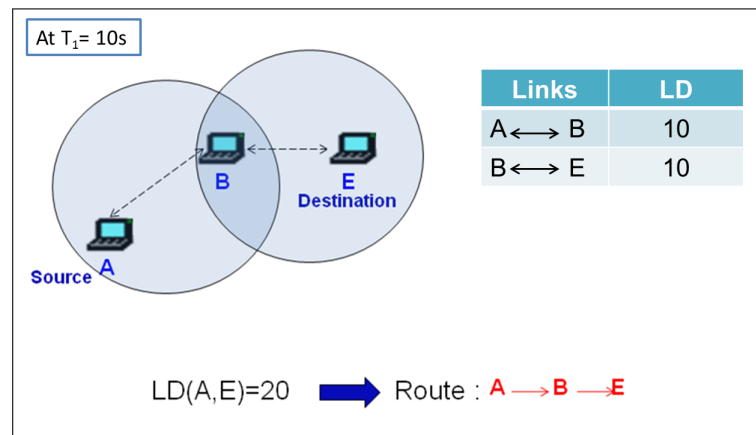
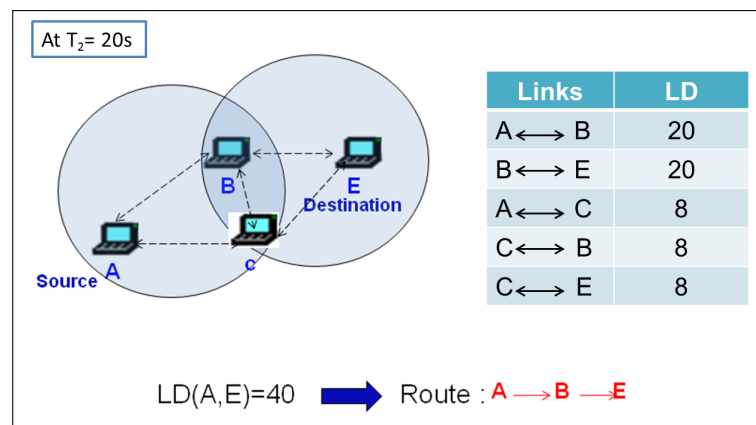
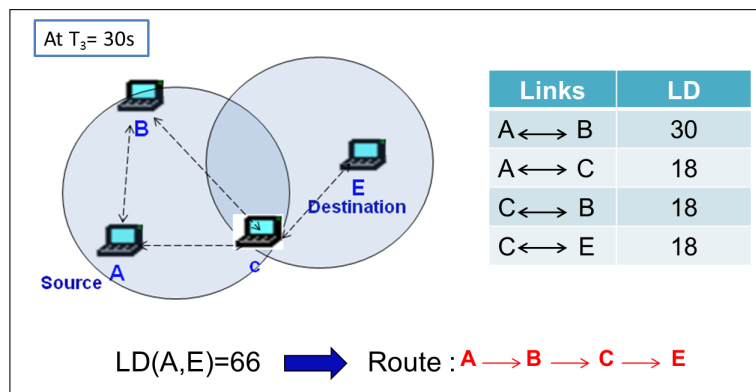
(a) Table de LD à un temps  $T=10s$ (b) Table de LD à un temps  $T=20s$ (c) Table de LD à un temps  $T=30s$ 

FIGURE 3.1 – Exemple de calcul de la métrique LD

À  $T = 12s$ , la topologie change : le nœud  $C$  entre en communication, alors des nouveaux liens symétriques se créent tel que  $(A, C)$ ,  $(B, C)$  et  $(C, E)$ . À cause de ces changements, la table de calcul de la métrique LD sera mise à jours selon les modifications apportées à la topologie.

À  $T = 20s$ , nous remarquons que les liens  $(A, B)$  et  $(B, E)$  persistent encore, alors  $LD(A, B)$  et  $LD(B, E)$  vont être incrémentés et prendre la valeur 20. Et LD pour les nouveaux liens sera mis à leur durée de vie qui est égale à  $8s$ , comme l'illustre la figure 3.1(b).

À  $T = 22s$ , la topologie change encore une fois : le nœud B bouge de manière que le lien entre B et E disparaît, car E n'est plus à la portée radio de B. Alors, la table de LD va être mise à jour selon ces changements.

À  $T = 30s$ , tous les liens persistent, donc LD pour chaque lien sera incrémentée selon la durée de vie de chaque lien, sauf (B, E) qui a disparu (voir figure 3.1(c)).

La durée d'une route (LD d'une route) est la somme des LD des différents sauts formant cette route. LD choisit les routes avec la plus longue durée de vie en supposant qu'ils sont les plus stables en cas de mobilité. Dans la figure 3.1, nous remarquons que la route choisie entre la source A et la destination E, est une route avec la plus longue durée de vie.

### 3.3 Topologies de référence

Afin d'effectuer l'évaluation des différentes métriques, trois scénarios sont considérés. Le premier scénario est une chaîne de 11 nœuds statiques avec un nœud mobile parcourant cette chaîne. Le deuxième scénario est un réseau mesh de 15 nœuds dont 9 nœuds sont posés en grille de  $3 \times 3$  et le reste des nœuds sont mobiles. Le dernier scénario présente un réseau ad hoc de 25 nœuds où tous les nœuds sont mobiles.

Ces trois topologies reflètent des exemples d'applications utilisées dans la vie quotidienne. La topologie chaîne peut être vue comme un exemple d'un réseau ad hoc véhiculaire (VANET) où un véhicule mobile parcourt une route munie de stations de base fixes. La topologie mesh peut être considérée comme un exemple de déploiement pour la couverture d'une zone blanche ou d'un réseau temporaire. La dernière topologie qui définit un réseau ad hoc où tous les nœuds sont mobiles peut être considéré comme un exemple d'un réseau militaire avec des soldats mobiles dans un champ de bataille. Nous expliquons ces trois topologies en détail dans le paragraphe suivant.

#### 3.3.1 Topologie chaîne

La première topologie considérée est une chaîne de 11 nœuds qui sont séparés par une distance de 100 mètres (voir figure 3.2). Un autre nœud mobile se déplace à vitesse constante le long de la chaîne des nœuds fixes. Cette topologie est un cas trivial qui nous permet de tester les performances des métriques, où les résultats sont faciles à interpréter. La source est le nœud mobile. Elle transmet des paquets de données à intervalles réguliers (trafic CBR). La destination est le premier nœud dans la chaîne.

Nous faisons varier la vitesse du nœud mobile à partir de  $10km/h$  à  $70km/h$ . La mobilité de la source et l'application démarrent après quelques secondes pour laisser le temps au protocole

OLSR de remplir les tables de routage. La simulation s'arrête lorsque le nœud mobile arrive à la fin de la chaîne.

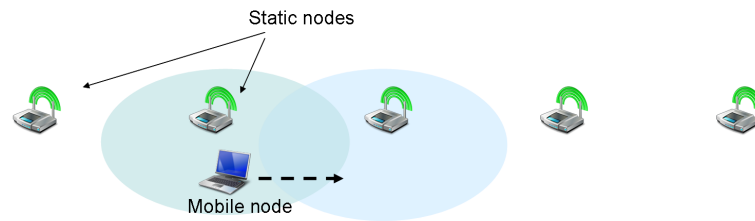


FIGURE 3.2 – Scénario 1 : un nœud mobile passe le long d'une chaîne fixe de nœuds à une vitesse constante. Il y a 12 nœuds : 11 nœuds statiques, chacun est placé à une distance de 100 mètres formant une chaîne, et un nœud mobile.

### 3.3.2 Topologie mesh

Dans le deuxième scénario, nous considérons une topologie maillée. Cette topologie dispose de 15 nœuds avec 6 nœuds mobiles et 9 nœuds fixes positionnés de manière à former une grille. Les nœuds mobiles suivent le modèle aléatoire de mobilité RWP [87]. Leur vitesse varie de  $10\text{km/h}$  à  $70\text{km/h}$ . Ce scénario est représenté dans la figure 3.3.

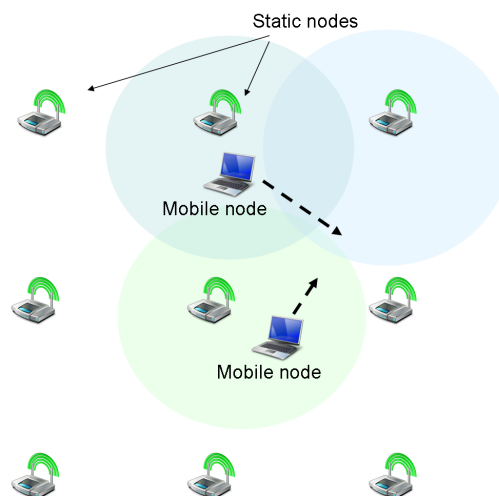


FIGURE 3.3 – Scénario 2 : un réseau maillé modélisé par une topologie réseau de 9 nœuds fixes formant une grille et 6 nœuds mobiles.

### 3.3.3 Topologie ad hoc

Dans le troisième scénario illustré dans la figure 3.4, nous considérons un réseau ad hoc, où tous les nœuds sont mobiles et sont répartis au hasard dans une zone de taille  $300 \times 300\text{m}^2$ . La mobilité des nœuds suit le modèle aléatoire RWP. La vitesse des nœuds mobiles varie entre  $10\text{km/h}$  à  $70\text{km/h}$ .

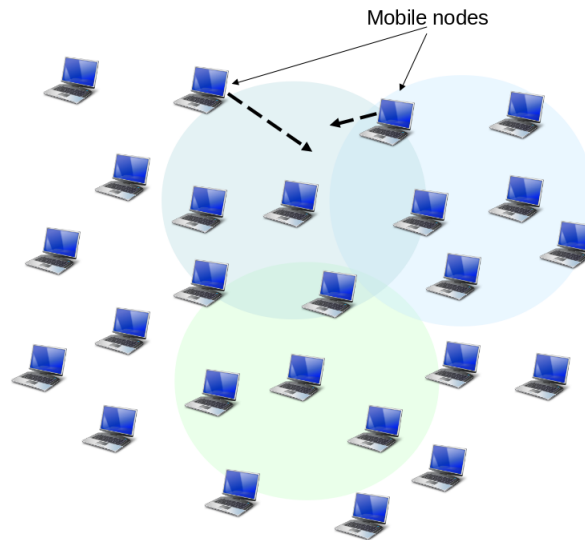


FIGURE 3.4 – Scénario 3 : réseau ad hoc de 25 nœuds mobiles

Parameters	Numerical Values
Simulation time	62 [sec]
Window size (Chain scenario)	$1200 \times 200 [m^2]$
Window size (Mesh and Ad hoc scenario)	$300 \times 300 [m^2]$
Number of nodes (Chain scenario)	12
Number of nodes (Mesh scenario)	15
Number of nodes (Ad hoc scenario)	25
Wireless technology	IEEE 802.11a
Traffic type	Constant Bit Rate (CBR)
Packet size	1024 bytes
WIFI_MANAGER	Minstrel_Wifi_Manager
HELLO_INTERVAL (OLSR)	0.25 sec
TC_INTERVAL (OLSR)	2 sec

TABLE 3.1 – Simulation Parameters

## 3.4 Évaluation des performances

### 3.4.1 Configuration de la simulation

Pour évaluer les différentes métriques, plusieurs simulations ont été effectuées à l'aide du simulateur NS-3, version 3.8 [91]. NS-3 est un simulateur réseau à événements discrets développés dans un but de recherche. Il présente l'un des outils de simulation les plus couramment utilisés et précis pour la recherche sur les réseaux [99]. Les paramètres utilisés pour les simulations sont donnés dans le tableau 3.1.

Afin de comparer les performances des différentes métriques de routage, nous avons besoin de critères de comparaison. Ces critères sont des métriques de performance qui dépendent des

critères recherchés par le protocole de routage. Dans notre étude, nous cherchons à maximiser le taux délivrance de paquets (PDR : Packet Delivery Ratio) ou de manière équivalente minimiser le taux de perte (Frame Error Rate), maximiser la capacité du réseau et réduire les délais de bout-en-bout. Nous utilisons donc le PDR, la capacité et le délai de bout-en-bout comme métriques de performance.

- **Taux de paquets livrés (Packet Delivery Ratio : PDR)** correspond au rapport entre le nombre de paquets de données reçus par la destination et le nombre de paquets de données envoyés par la source. Le PDR est calculé comme suit :

$$PDR = \frac{\sum Paquets\_recus}{\sum Paquets\_envoyes} \quad (.3.4.6)$$

- **Capacité** est la quantité du trafic envoyée par tous les nœuds (N) du réseau et écoulee pendant la durée de simulation. Cette métrique représente la quantité maximale de trafic que peut écouler le réseau. Une capacité plus importante du réseau permet d'offrir une meilleure qualité de service à un plus grand nombre d'utilisateurs [100]. Cette métrique est calculée de la façon suivante :

$$Capacite = \frac{\sum_{n \in N} |Paquets\_recus|}{Temps\_de\_l'application} \quad (.3.4.7)$$

- **Délai de bout-en-bout (End-to-End delay)** est le temps moyen entre l'émission d'un paquet et sa réception. Le délai de bout-en-bout comprend le retard de routage et d'autres retards divers, tels que, le retard de transmission, le retard de propagation et le retard de la file d'attente. Le délai de bout-en-bout est calculé de la manière suivante :

$$Delai = \frac{\sum_{n \in N} delai}{\sum Paquets\_recus} \quad (.3.4.8)$$

Nous avons mesuré les performances des différentes métriques : Hop Count, ETX, ETT et LD, afin de les comparer. Pour les scénarios de simulation, nous avons utilisé les trois topologies déjà décrites dans 3.3.

Dans les simulations, nous avons pris une seule source et une seule destination qui sont choisies au hasard parmi les nœuds mobiles. La source envoie deux paquets par seconde à la destination. Nous avons opté pour un trafic très léger afin d'éviter les problèmes de congestion qui mènent à des pertes de messages de contrôle et des routes obsolètes. Chaque point sur les courbes est la moyenne de 20 simulations et il est représenté avec un intervalle de confiance de 95%.

### 3.4.2 Résultats pour la chaîne de noeud

La topologie chaîne est constituée de 12 noeuds dont 11 sont fixes, séparés par une distance de 100 mètres, formant une chaîne et un noeud mobile parcourant cette chaîne. La vitesse du noeud

mobile varie 10km/h à 70km/h.

- **PDR**

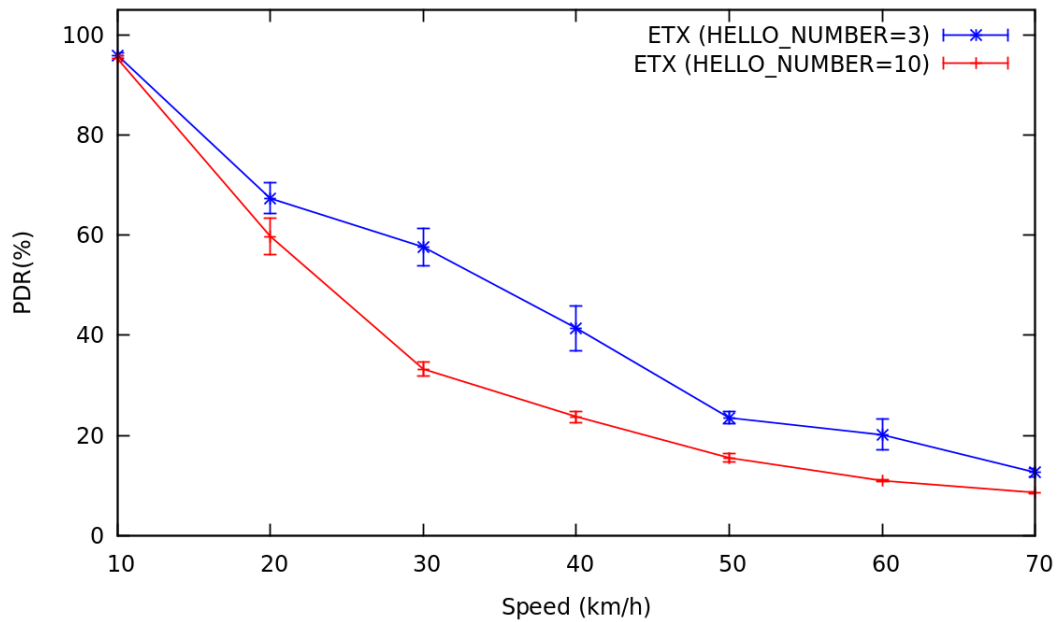


FIGURE 3.5 – PDR en fonction de la vitesse pour ETX avec `HELLO_NUMBER` = 3 et 10 (Chaîne de noeuds)

Dans ce scénario, le *PDR* est utilisé comme mesure de performance. Nous expliquons le choix de la variable *HELLO\_NUMBER*, avant d'évaluer les performances des différentes métriques en terme de *PDR*. La variable *HELLO\_NUMBER* utilisé dans le calcul d'ETX prend deux valeurs : soit 3, soit 10. Le premier choix (*HELLO\_NUMBER* = 3) désigne un petit intervalle d'historique qui sert à prendre que les nouvelles valeurs d'ETX de cette manière le protocole gère mieux les changements brusques de l'environnement tel que la mobilité des nœuds. Par contre, le deuxième choix désigne un intervalle d'historique plus large qui lisse les changements brusques mais il est moins réactif.

La figure 3.5 montre qu'ETX avec *HELLO\_NUMBER* = 3 donne des meilleures performances qu'ETX avec *HELLO\_NUMBER* = 10, cela s'explique par le fait qu'un petit historique reflète mieux l'état de l'environnement qu'un grand historique de message HELLO. Pour cette raison, nous avons pris *HELLO\_NUMBER* = 3 pour le reste des simulations.

La figure 3.6 montre le *PDR* pour les différentes métriques en faisant varier la vitesse du nœud mobile. Nous observons que le *PDR* des métriques se dégrade en fonction de la vitesse, principalement en raison du délai entre les valeurs prises en compte par le protocole de routage et les valeurs réelles des métriques. Les performances des deux métriques ETX et ETT sont meilleures que Hop Count et LD car ETX et ETT privilégient les routes avec moins de pertes et à hautes capacités. LD est la pire métrique car elle choisit des chemins



avec la plus longue durée. En effet, dans ce scénario, la métrique LD n'est pas pertinente car les nouveaux liens doivent être privilégiés plutôt que les anciens.

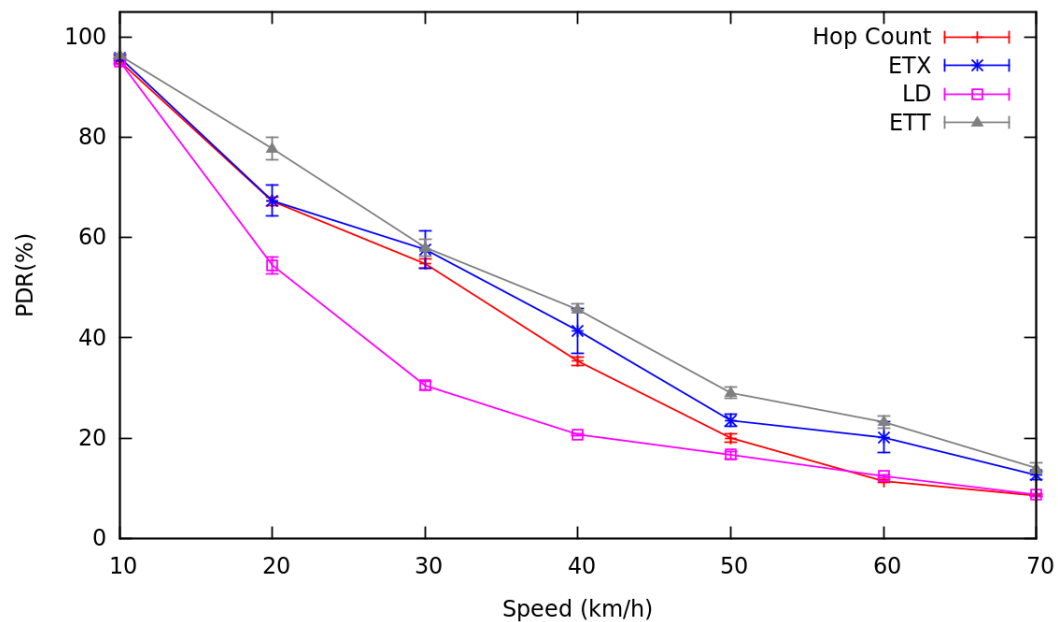


FIGURE 3.6 – Comparaison du PDR en fonction de la vitesse pour Hop Count, ETX et ETT (Chaîne de noeuds)

- **Capacité** Nous avons tracé le débit exprimé en  $Kb/s$  en fonction du débit de la source exprimé en  $Kb/s$ , pour une vitesse fixe de  $25km/h$ . Dans ce scénario, nous avons pris une source et une destination. La figure 3.7 illustre la capacité des métriques Hop Count, ETX et ETT, pour la topologie chaîne. Nous observons dans cette figure, que les performances d'ETT et d'ETX sont meilleures que celle de Hop Count, car ils choisissent les routes à hautes capacités et avec moins de pertes. Mais, les trois métriques atteignent un débit de  $680Kb/s$ .
- **Délai de bout-en-bout** La figure 3.8 s'intéresse à l'évaluation du délai en fonction de la vitesse du nœud mobile. Nous constatons que le délai des différentes métriques augmentent en fonction de la vitesse. Les métriques ETX et ETT peuvent avoir un délai de propagation plus important que Hop Count parce que certains paquets sont transmis par des chemins plus longs. Pour cette raison, le délai d'OLSR est moins long que le délai d'ETX et d'ETT, malgré qu'elles donnent des meilleures performances en terme de PDR que la métrique Hop Count.

Après avoir montré les résultats de simulation de la topologie chaîne avec les différentes métriques de performance, nous présentons par la suite les résultats pour le réseau mesh.

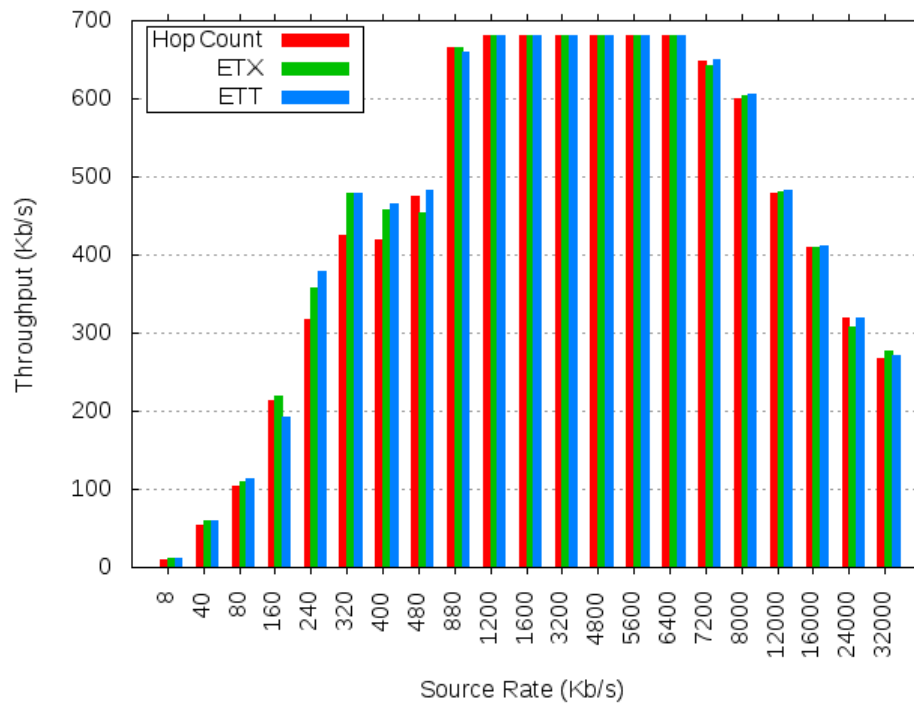


FIGURE 3.7 – Capacité des métriques Hop Count, ETX et ETT en fonction du débit de la source (Chaîne de noeuds)

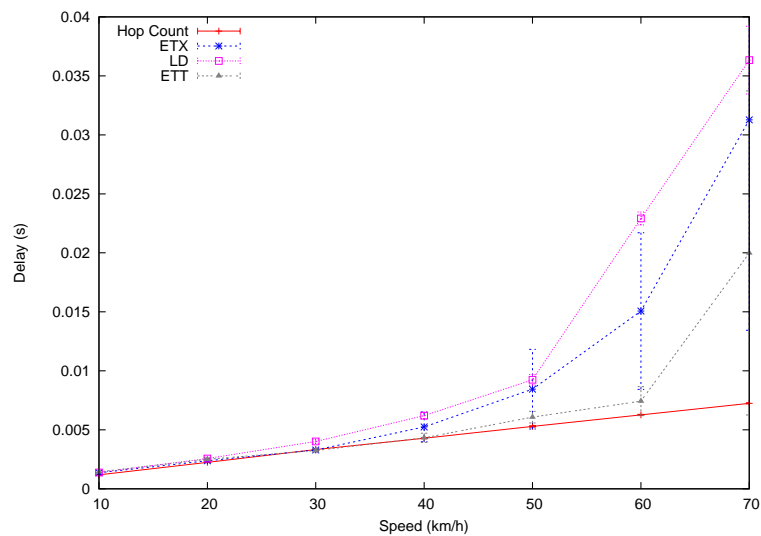


FIGURE 3.8 – Délai de bout-en-bout en fonction de la vitesse pour Hop Count, ETX et ETT (Chaîne de noeuds)

### 3.4.3 Résultats pour le réseau mesh

Cette topologie est constituée de 15 nœuds : 9 nœuds fixes placés en grille de  $3 \times 3$ , et 6 nœuds mobiles suivant le modèle de mobilité RWP. La vitesse des nœuds mobiles varie entre  $10\text{km/h}$  et  $70\text{km/h}$ .

- **PDR** Nous commençons par comparer ETX avec *HELLO\_NUMBER* = 3 et ETX avec *HELLO\_NUMBER* = 10, en terme de PDR. Nous observons dans la figure 3.9, que les deux courbes se croisent et sont presque similaires. Pour ETX, le PDR est meilleur lorsque l'historique utilise un *HELLO\_NUMBER* de 10 et que la vitesse est inférieure à 22km/h. Mais au delà de cette vitesse, le PDR pour *HELLO\_NUMBER* égal à 3 est plus élevé que pour *HELLO\_NUMBER* égal à 10. Cela s'explique par le fait que l'utilisation d'un petit historique de message HELLO gère mieux la mobilité.

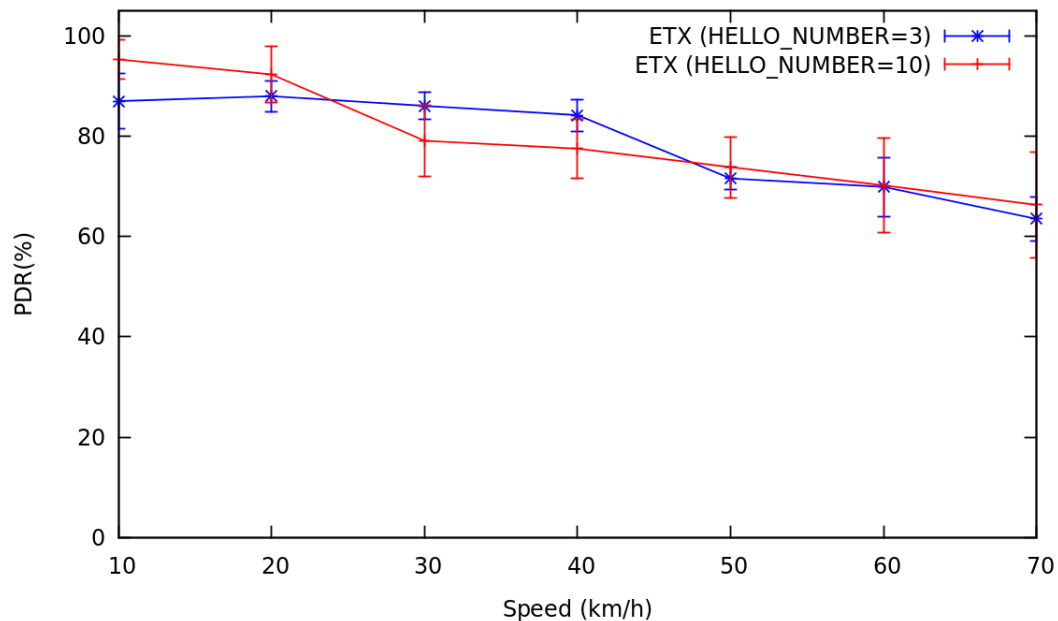


FIGURE 3.9 – PDR en fonction de la vitesse pour ETX avec *HELLO\_NUMBER* = 3 et 10 (Mesh)

La figure 3.10 montre le PDR des différentes métriques en fonction de la vitesse des nœuds mobiles. Nous observons que les performances d'ETT et d'ETX sont meilleures que Hop Count et LD car ils prennent en considération le taux de perte et le débit des liens. Dans ce scénario, LD reste aussi la pire métrique en terme de PDR parce qu'elle choisit les routes avec la plus longue durée de vie.

#### - Capacité

Nous avons tracé le débit en fonction du débit de la source comme l'illustre la figure 3.11, avec une vitesse de 25km/h pour les nœuds mobiles. Dans cette figure, nous observons que la capacité d'ETT est meilleure que celle d'ETX et de Hop Count. En effet, ETT choisit les routes avec le moins taux de pertes et avec le meilleur débit. Le débit d'ETT atteint 615Kb/s et celui d'ETX atteint 607Kb/s. C'est donc très similaire. Le débit choisi par Minstrel dépend du taux de perte des différentes vitesses de transmission. ETX et ETT

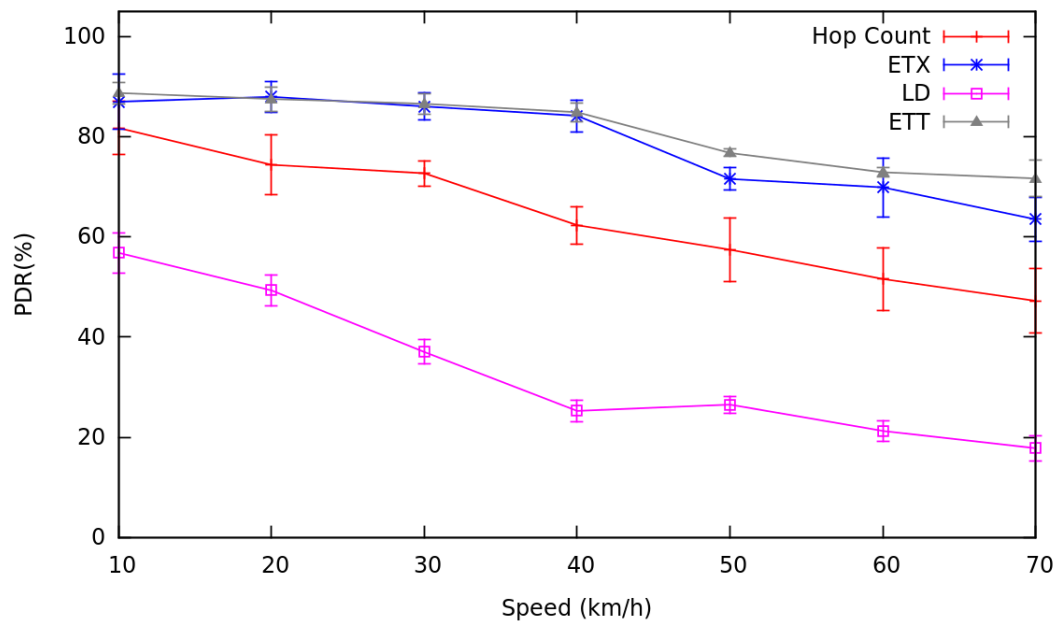


FIGURE 3.10 – Comparaison du PDR en fonction de la vitesse pour Hop Count, ETX et ETT (Mesh)

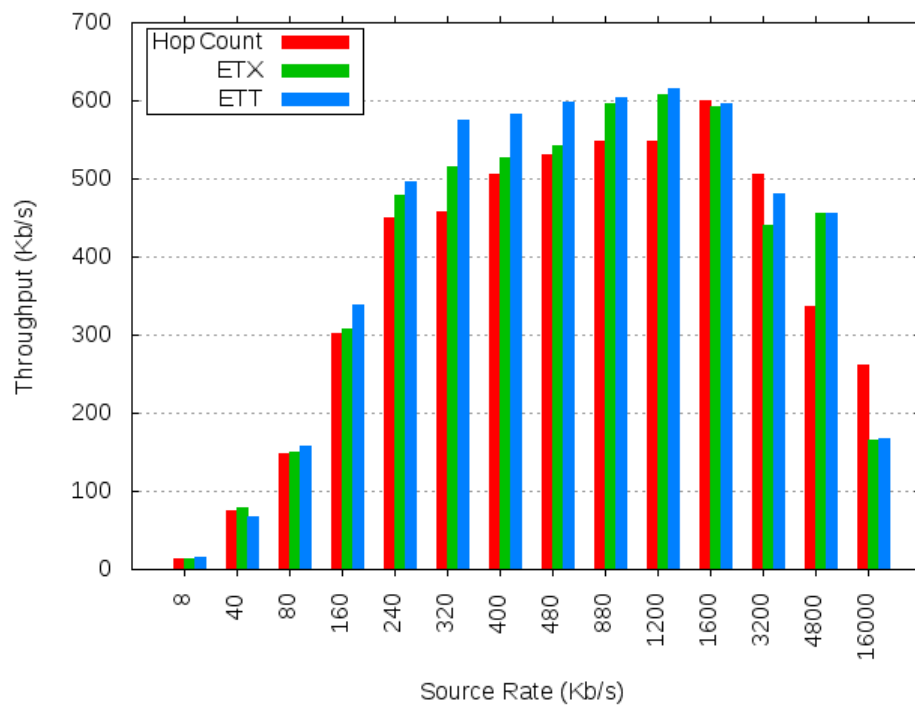


FIGURE 3.11 – Capacité des métriques Hop Count, ETX et ETT en fonction du débit de la source (Mesh)

sont donc assez liés. Les performances d'ETX sont moins bonnes que celles d'ETT car elle privilège les routes avec juste moins de pertes sans prendre en considération le débit des liens. Mais la capacité d'ETX reste meilleure que la capacité de Hop Count. En effet, Hop Count est la pire métrique dans ce cas, car elle choisit les routes avec le minimum de

sauts sans prendre en considération la qualité des liens.

#### - Délai de bout-en-bout

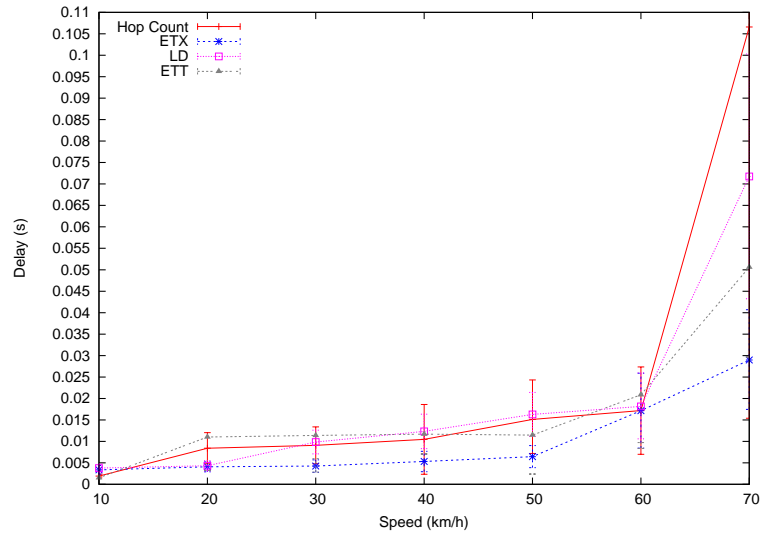


FIGURE 3.12 – Délai de bout-en-bout en fonction de la vitesse pour Hop Count, ETX et ETT (Mesh)

Pour ce scénario, nous traçons aussi le délai de bout-en-bout des différentes métriques dans la figure 3.12. Nous remarquons que Hop Count et LD présentent des délais plus importants que les délais d'ETX et d'ETT. Le délai de bout-en-bout comprend le délai de transmission, le délai de propagation à partir de la source vers la destination et du délai de file d'attente dans les nœuds relais. Hop Count et LD ont un temps de propagation plus important qu'ETX et ETT car le délai comporte le délai de transmission. Selon la figure 3.10, Hop Count et LD présentent un taux de perte de paquet plus important par rapport à ETX et ETT. Alors, il y aura retransmission des paquets ce qui engendre un délai supplémentaire de transmission pour Hop Count et LD.

Dans ce paragraphe nous nous sommes intéressés aux résultats de simulation du réseau mesh. Par la suite, nous montrons les résultats pour la topologie aléatoire.

#### 3.4.4 Résultats pour la topologie aléatoire

Nous considérons 25 nœuds mobiles suivant le modèle de mobilité RWP et répartis d'une manière aléatoire dans une zone de taille  $300 \times 300m^2$ . Les nœuds forment une grille de  $5 \times 5$ .

#### - PDR

Nous comparons les performances en terme de PDR d'ETX avec les deux valeurs de *HELLO\_NUMBER*, afin de déterminer lequel des deux choix est plus précis. La figure

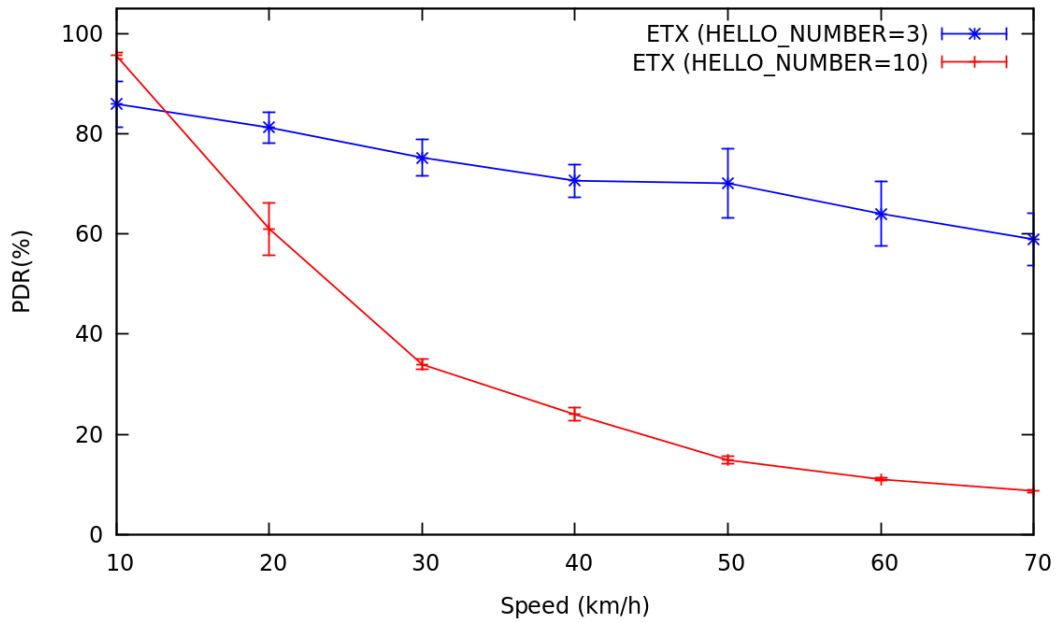


FIGURE 3.13 – PDR en fonction de la vitesse pour ETX avec *HELLO\_NUMBER* = 3 et 10 (Topologie aléatoire)

3.13 reflètent le PDR d'ETX avec les deux valeurs de *HELLO\_NUMBER*. Nous observons que le PDR de ETX avec *HELLO\_NUMBER* = 10 surpasse celui d'ETX avec *HELLO\_NUMBER* = 3 pour des vitesses inférieures à 15km/h. Par contre, pour des vitesses supérieures à 15km/h, la situation s'inverse : PDR d'ETX avec *HELLO\_NUMBER* = 3 surpasse largement le PDR d'ETX avec *HELLO\_NUMBER* = 10. Cela s'explique par le fait que le choix de *HELLO\_NUMBER* = 3 est plus réactif.

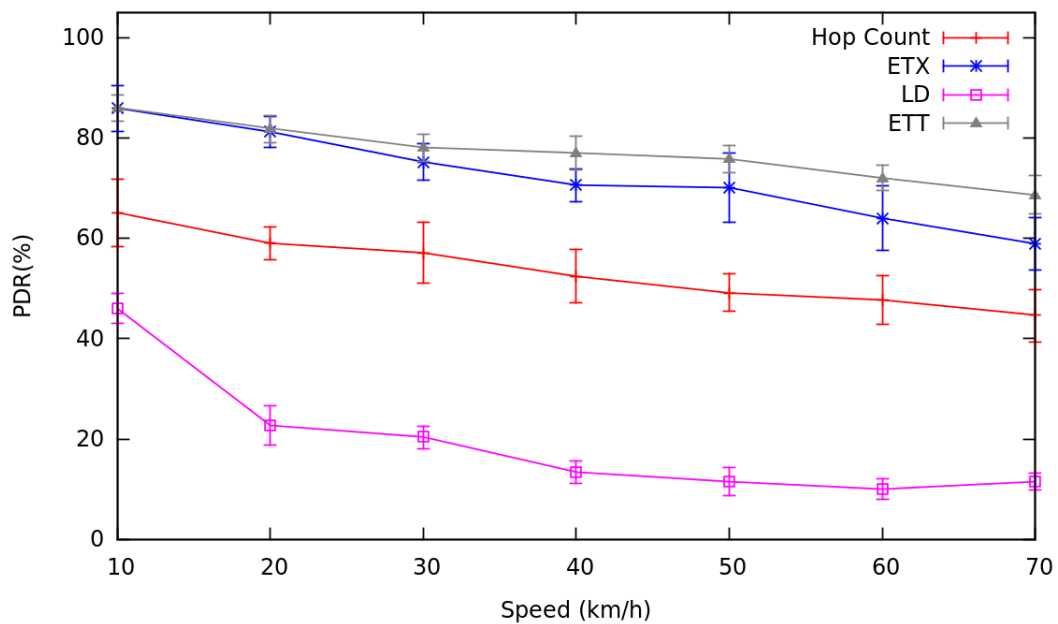


FIGURE 3.14 – Comparaison du PDR en fonction de la vitesse pour Hop Count, ETX et ETT (Topologie aléatoire)

Nous observons que le PDR des trois métriques se dégrade en fonction de la vitesse. La figure 3.14 montre que les performances d'ETT et d'ETX restent meilleurs en terme de PDR que celle de Hop Count même dans ce scénario. En effet, ETX et ETT privilègent les routes avec moins de pertes, et en plus ETT choisit les routes à hautes capacités.

#### - Capacité

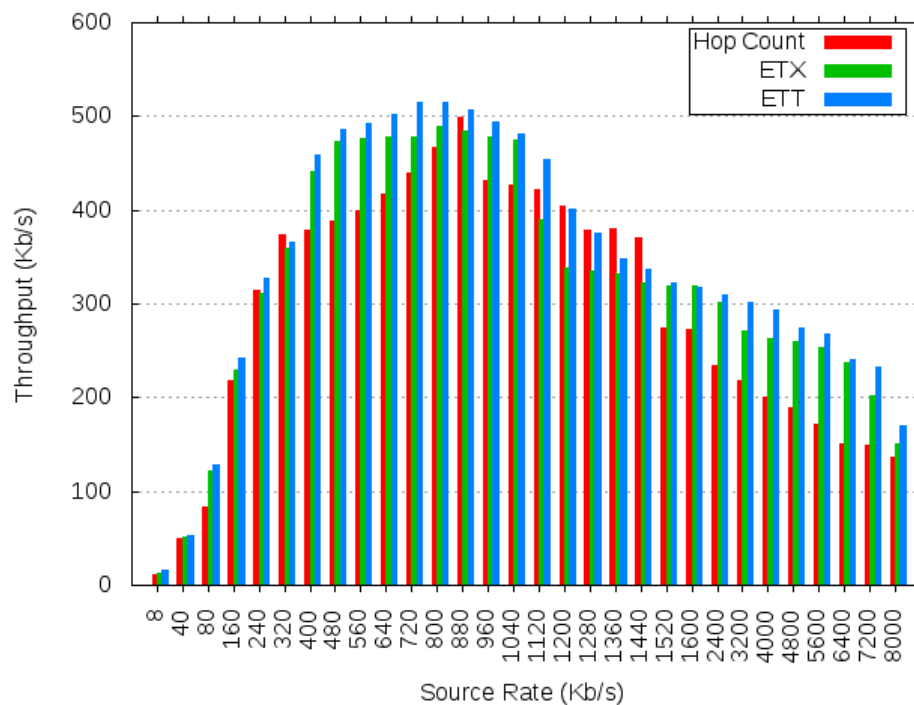


FIGURE 3.15 – Capacité des métriques Hop Count, ETX et ETT en fonction du débit de la source (Topologie aléatoire)

Nous mesurons aussi la capacité des métriques classiques pour la topologie ad hoc. La vitesse des nœuds mobiles est fixée à 25km/h. La figure 3.15 montre le débit en fonction du débit de la source. Nous remarquons que le débit d'ETT surpasse celui d'ETX et de Hop Count : il dépasse 515Kb/s.

#### - Délai de bout-en-bout

La figure 3.16 comporte le délai de bout-en-bout des différentes métriques en fonction de la vitesse des nœuds. Encore une fois, nous observons qu'ETX et ETT présentent des délais moins importants que Hop Count et LD. Cela s'explique par le taux de perte significatif de Hop Count et LD par rapport à ETX et ETT se qui engendre un délai plus important pour la retransmission des paquets perdus d'où une augmentation dans le délai de bout-en-bout.

Au cours de cette section, nous avons montré une évaluation des performances des différentes métriques de routage classiques pour les trois topologies proposées : chaîne, mesh et aléatoire.

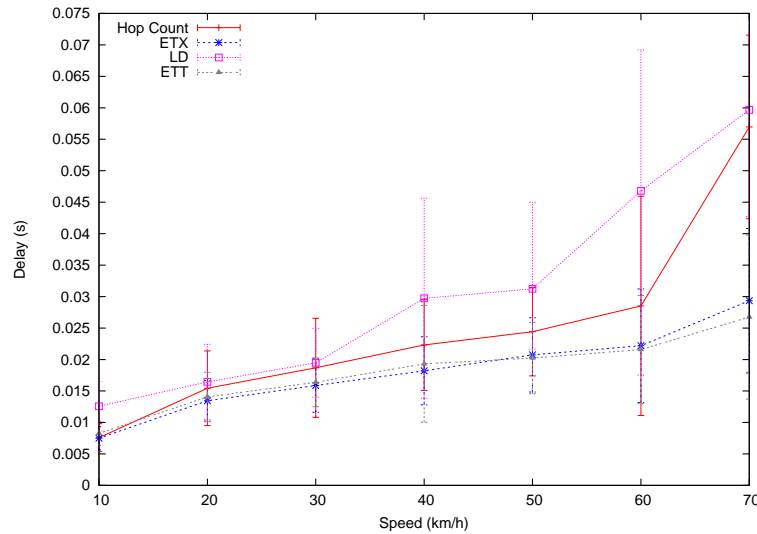


FIGURE 3.16 – Délai de bout-en-bout en fonction de la vitesse pour Hop Count, ETX et ETT (Topologie aléatoire)

Cette évaluation est basée sur des mesures des métriques de performance telle que : le PDR, la capacité et le délai de bout-en-bout.

### 3.5 Conclusion

Dans ce chapitre, nous avons présenté notre approche de mise en oeuvre d'un ensemble de métriques de routage dans OLSR. La première métrique de routage implémentée est ETX. Nous avons utilisé à la base une fenêtre glissante pour stocker l'historique des messages HELLO (reçu ou perdu), pour chaque lien sans fil. Cette fenêtre nous permet de calculer les deux valeurs  $d_f$  et  $d_r$  qui prévoit la valeur d'ETX. À partir de cette valeur, nous pouvons en déduire la valeur de la métrique ETT, en estimant la valeur du débit échangé entre deux nœuds à partir du *WIFI\_MANAGER*. La dernière métrique mise en oeuvre est la métrique LD. Elle présente la durée de vie d'une liaison entre deux nœuds.

Plusieurs scénarios de simulation ont été effectués, afin d'évaluer les métriques Hop Count, ETX, ETT et LD. Les résultats de simulation montrent qu'ETX et ETT sont meilleures que Hop Count et LD en terme de PDR, de capacité et du délai de bout-en-bout. Cela se traduit par une sélection de route plus pertinente (avec moins de pertes et avec un meilleur débit) pour ETX et ETT par rapport à Hop Count et LD.

Bien que les résultats de simulation montrent qu'ETX et ETT sont meilleures que Hop Count et LD, elles présentent des taux de perte importants lorsque les nœuds bougent. Ces métriques restent incapables de gérer la mobilité car il existe un décalage entre les valeurs réelles des métriques et celles prises en compte dans le processus de calcul de route. Ces métriques doivent



donc prendre en compte plus de critères de qualité de lien pour mieux gérer la mobilité et réduire (ou supprimer) ce décalage.

## Chapitre 4

# Algorithmes d'anticipation des métriques de routage

### Sommaire

---

<b>4.1 Algorithmes d'anticipation des métriques de routage</b>	<b>90</b>
4.1.1 Anticipation basée sur le développement de Taylor	91
4.1.1.1 Paramètres de Conception	93
4.1.2 Anticipation basée sur la prédiction de la puissance du signal reçu	94
4.1.2.1 Relations entre FER et la puissance du signal	95
4.1.2.2 Anticipation de la métrique	96
4.1.2.3 Algorithme de prédiction de la puissance du signal	96
4.1.2.4 Méthode de régression linéaire	97
4.1.2.4.1 Expérimentations	99
4.1.2.5 Configuration des paramètres	99
<b>4.2 Évaluation des Performances</b>	<b>100</b>
4.2.1 Topologie chaîne	102
4.2.1.1 PDR	102
4.2.1.2 Capacité	105
4.2.1.3 Délai de bout-en-bout	106
4.2.2 Topologie mesh	109
4.2.2.1 PDR	109
4.2.2.2 Capacité	112
4.2.2.3 Délai de bout-en-bout	112
4.2.3 Topologie aléatoire	113
4.2.3.1 PDR	115
4.2.3.2 Capacité	117
4.2.3.3 Délai de bout-en-bout	119

---

**4.3 Conclusion . . . . . 120**


---

Dans ce chapitre, nous proposons des métriques sensibles à la mobilité et qui reflètent la qualité du lien en même temps. Les métriques ETX et ETT sont efficaces pour trouver des chemins qui optimisent le débit du réseau. En effet, ces métriques donnent des valeurs pertinentes sur la qualité des liens [53]. Malheureusement, elles ne sont pas efficaces pour gérer la mobilité, car il y a un délai généré entre leurs valeurs réelles/instantanées et celles prises en compte dans le processus de calcul de route. Par conséquent, nous proposons d'anticiper les valeurs de ces métriques quelques secondes à l'avance. Dans ce chapitre, nous présentons notre approche de calcul de la métrique ETX anticipée et ETT anticipée notée ETX\_ANT et ETT\_ANT respectivement dans le reste du document.

Nous proposons deux algorithmes pour anticiper la métrique ETX (respectivement ETT). Le premier algorithme d'anticipation est basé sur un simple développement de Taylor. Par contre, le deuxième algorithme d'anticipation est basé sur un algorithme de prédiction de la puissance du signal. Nous expliquons les deux algorithmes ainsi que les paramètres utilisés pour leur conception. Nous présentons aussi les résultats de simulation. Nous nous sommes intéressés ainsi à une comparaison de ces deux approches afin d'évaluer leurs performances, pour en déduire l'algorithme le plus performant pour la gestion de la mobilité.

## 4.1 Algorithmes d'anticipation des métriques de routage

Les approches de prédiction de la mobilité peuvent détecter la prochaine station de base pour la réservation de ressources avant le handover. Cette approche a attiré plusieurs intérêts de recherche [101], [102], [103], [104]. L'intégration du transfert intercellulaire dans les couches 2 et 3 a été proposé dans [105]. Les auteurs décrivent une approche de transfert intercellulaire prédictive pour IP Mobile (Mobile IP) pour supporter un transfert rapide ou lisse. Ils utilisent une estimation de la mobilité en temps réel pour prédire un futur transfert.

Dans [106], l'indicateur de puissance du signal reçu (RSSI) et la direction du nœud mobile sont également utilisés comme paramètres d'entrée à un système d'inférence floue pour prédire la décision du handover. Le papier présente les approches Mobility and Signal Strength-Aware Hand-off Decision (MSSHD) pour prédire la décision du handover.

Une nouvelle machine d'apprentissage basée sur un système de prédiction est considérée dans [107]. Les auteurs utilisent les informations disponibles dans les réseaux cellulaires comme Channel State Information (CSI) et l'historique du handover. Ils intègrent Support Vector Machines (SVM) dans une structure efficace de prétraitement pour résoudre le problème de prédiction.

Tous ces mécanismes ont été conçus pour les réseaux cellulaires ou les réseaux d'infrastructure, où un mobile ne doit sélectionner que le meilleur lien (associé au meilleur point d'infrastructure). Ces techniques ne sont donc pas appropriées dans les réseaux ad hoc où les liens peuvent être potentiellement utilisés.

En outre, de nombreuses approches utilisant des algorithmes de prédiction de la mobilité ont été proposées dans le contexte des réseaux ad hoc. Une prédiction de la mobilité basée sur un réseau de neurones et un enregistrement de la localisation géographique des nœuds est introduit dans [108]. Dans [109], une méthode de prédiction de la mobilité adaptative est proposée. Cette méthode prédit la future distance entre 2 nœuds voisins à l'aide des automates. Une approche de prédiction de la mobilité basée sur un protocole de message HELLO est développée dans [110]. Chaque nœud et l'ensemble de ses voisins prédisent leurs propres positions au moyen d'un modèle de mobilité basé sur l'autorégression.

Toutes ces techniques ont été conçues pour les réseaux ad hoc dans le but de prédire les lieux des futurs nœuds ou la distance entre eux. Mais en général, les nœuds ne connaissent pas leurs positions car ils ne sont pas équipés de dispositifs de géolocalisation, et la mobilité est souvent imprévisible (en raison des changements de direction, par exemple). En outre, les variations des qualités de liaison ne sont pas seulement liées à la distance entre les nœuds, mais aussi à l'environnement dans lequel évoluent les nœuds (ville, construction, etc.) et les phénomènes complexes de la radio (décoloration, ombrage, etc). Par conséquent, nous pensons qu'une méthode de prédiction de la qualité de liaison plus appropriée doit être fondée sur la puissance du signal plutôt que des lieux géographiques.

Dans le paragraphe suivant, nous présentons les deux algorithmes proposés dans le but d'anticiper les métriques de routage.

#### 4.1.1 Algorithme 1 : Anticipation basée sur le développement de Taylor

L'idée de base de notre algorithme est d'utiliser le calcul d'ETX classique lorsque la qualité du lien est bonne, et d'augmenter artificiellement et progressivement la métrique lorsque la qualité du lien se dégrade. Chaque nœud calcule la métrique anticipée pour chaque lien (chaque voisin). Nous supposons que les nœuds ont un tableau reliant la qualité du lien mesurée à la couche physique et le FER (Frame Error Rate). Cette quantité doit refléter la qualité du lien. Elle peut être soit : le rapport signal à bruit (SNR), l'Indicateur de Force du Signal Reçu (RSSI) souvent disponible pour 802.11, les cartes 3G/LTE, ou toute autre quantité qui reflète la qualité du lien. Cette quantité peut être fournie par la carte elle-même ou estimée par les nœuds quand elle n'est pas disponible.

**Algorithm 3:** Management Mobility ALGORITHM

---

```

void ETXcalculation()
begin
    /* The different variables are updated at each HELLO reception */
    Update  $LINK_Q$  with the last Hello measurement;
    Update  $CURRENT\_TIME$  with the time of this measurement;
     $RATE = \frac{LINK_Q - OLD\_LINK_Q}{TIME}$ ;
     $TIME = CURRENT\_TIME - OLD\_TIME$ ;

    if  $LINK_Q > TH_Q$  OR  $RATE > 0$  then
        /* The link quality is good or is improving: we do not anticipate */
         $ETX\_ANT = UPDATE(ETX)$ ;
    else
        /* The quality is under the  $TH_Q$  threshold: we anticipate */
         $SECOND\_RATE = \frac{RATE - OLD\_RATE}{TIME}$ ;
         $ANT\_LINK_Q =$ 
         $LINK_Q + RATE \times ANTICIPATION\_TIME + SECOND\_RATE * ANTICIPATION\_TIME^2$ ;
         $ETX\_ANT = ETX\_ANT(ANT\_LINK_Q)$ ;
     $OLD\_RATE = RATE$ ;
     $OLD\_LINK_Q = LINK_Q$ ;
     $OLD\_TIME = CURRENT\_TIME$ ;

```

---

L'algorithme est résumé dans l'algorithme 3. Il est effectué localement pour chaque liaison sans fil. Les variables suivantes sont utilisées dans l'algorithme :

- $LINK_Q$  : la mesure courante de la qualité de la liaison.
- $TH_Q$  : le seuil sur la qualité de la liaison. L'anticipation est réalisée uniquement lorsque la qualité de la liaison est inférieure à ce seuil.
- $OLD\_LINK_Q$  : la mesure de qualité précédente.
- $CURRENT\_TIME$  : l'instant de la mesure actuelle.
- $OLD\_TIME$  : le temps de la mesure précédente.
- $TIME$  : le temps entre les deux dernières mesures.
- $RATE$  : pente de la qualité de la liaison.
- $OLD\_RATE$  : pente précédente.
- $SECOND\_RATE$  : pente de  $RATE$ , il s'agit d'une estimation approximative de la dérivée seconde.

Lorsqu'un nœud reçoit un paquet d'un voisin (un message HELLO), la métrique associée à ce lien est mise à jour. Nous utilisons un seuil sur la qualité du lien notée  $TH_Q$ . Si la qualité du lien mesurée notée  $LINK_Q$  pour ce paquet est supérieure à  $TH_Q$ , le lien a une qualité suffisante. Dans ce cas, il n'est pas nécessaire d'anticiper la métrique ETX. La métrique associée à ce lien est alors la métrique ETX classique. Elle est mise à jour avec la méthode classique notée  $UPDATE(ETX)$  dans notre algorithme.

Si  $LINK_Q$  est inférieur à  $TH_Q$ , nous avons deux cas possibles. Lorsque  $LINK_Q$  s'améliore, c'est-à-dire si la variable  $RATE$  est positive, il n'est pas nécessaire d'anticiper la métrique et nous utilisons à nouveau la méthode classique. Si la qualité de la liaison diminue ( $RATE \leq 0$ ), nous anticipons la métrique  $ANTICIPATION\_TIME$  secondes à l'avance par une projection sur la base de l'estimation de la dérivée première et la dérivée seconde de  $LINK_Q$ . Nous utilisons cette dérivée seconde parce que  $LINK_Q$  ne se comporte pas comme une fonction linéaire. En effet, l'environnement radio est complexe, et la puissance du signal ou le SINR sont rarement linéaires en fonction du temps. Par conséquent, nous pouvons voir notre méthode d'anticipation comme un développement simple de Taylor. Ensuite, nous associons la qualité du lien anticipée  $ANT\_LINK_Q$  à la valeur  $ETX\_ANT$  correspondante. Elle est réalisée par la fonction  $ETX\_ANT(.)$ . Cette fonction utilise le mapping prédéfini entre les valeurs de  $LINK\_Q$  et les valeurs d' $ETX$ . Ce mapping peut être calculé à partir du modèle radio ou à partir d'un ensemble de mesures. Dans la section suivante, nous décrivons la méthode que nous avons mise en place pour paramétrer notre algorithme [111].

#### 4.1.1.1 Paramètres de Conception

Dans cette section, nous définissons les règles de conception de notre métrique. Plus précisément, nous présentons la méthode utilisée pour calculer  $TH_Q$ . Nous supposons que nous connaissons la relation entre la qualité de la liaison et le FER. Dans notre implémentation, nous considérons que notre qualité de lien notée  $LINK_Q$  est le SNR, et il est facile d'avoir des valeurs approximatives de FER en fonction du SNR. Dans la Fig 4.1, nous avons tracé la fonction du FER par rapport au SNR. Elle a été effectuée avec le simulateur NS-3 [91]. Nous utilisons cette fonction, notée  $FER(SNR)$ , pour mapper FER et SNR. Pour une mise en œuvre réelle, un mapping équivalent peut être obtenu à partir d'un ensemble d'expérimentations.

Dans nos simulations, notre qualité de lien est la même dans les deux sens, il est donc facile de dériver les valeurs ETX à partir du SNR :

$$ETX\_ANT(SNR) = \frac{1}{FER(SNR)^2} \quad (4.1.1)$$

Afin d'avoir une estimation des valeurs d'ETX plus précises dans un déploiement réel,  $d_f$  et  $d_r$  doivent être échangés dans les paquets HELLO. L'autre paramètre à régler est le seuil  $TH_Q$  qui déclenche le processus d'anticipation.

Nous observons dans la figure 4.1, que le FER augmente brusquement et approximativement de 10% à 100% lorsque le SNR est inférieur à 1.11. Ce processus signifie que si le lien se dégrade à cause de la mobilité des nœuds, le FER augmente très rapidement. La métrique ETX classique ne sera pas mise à jour suffisamment rapidement pour considérer le lien en question comme un lien avec perte par le protocole de routage, et les paquets seront perdus. Nous supposons que le

temps considéré pour prendre en compte une nouvelle métrique par le protocole de routage est donné par la variable TIME. Par conséquent, FER augmente fortement quand SNR est inférieure à une valeur donnée notée  $SNR_{th}$  (environ 1.11 sur la figure 4.1). Alors, les mesures d'ETX sont plus ou moins constantes lorsque le SNR est supérieur à  $SNR_{th}$  et augmente soudainement pour des valeurs inférieures.

L'idée de base est donc d'anticiper ETX à l'égard de  $SNR_{th}$  : nous anticipons ETX ANTICIPATION\_TIME secondes à l'avance à l'égard du temps où le SNR atteint  $SNR_{th}$ . Si nous supposons que la vitesse relative des nœuds est connue, notée  $S_{rel}$ , et donc la fonction d'atténuation, notée  $PL(.)$ , alors nous pouvons facilement déduire  $TH_Q$  comme suit :

$$TH_Q = \frac{P_t \times PL(d_{SNR} + S_{rel} \times ANTICIPATION\_TIME)}{N} \quad (4.1.2)$$

Où  $P_t$  est la puissance de transmission,  $d_{SNR}$  est la distance à laquelle le SNR est égal à  $SNR_{th}$  ( $SNR_{th} = \frac{P_t PL(d_{SNR})}{N}$ ), et  $N$  est le bruit. Dans la figure 4.2, nous traçons la métrique classique ETX et la métrique anticipée ETX\_ANT en fonction du SNR. Nous observons qu'ETX\_ANT est tout simplement un décalage de la métrique ETX classique équivalent à ANTICIPATION\_TIME seconde à l'avance.

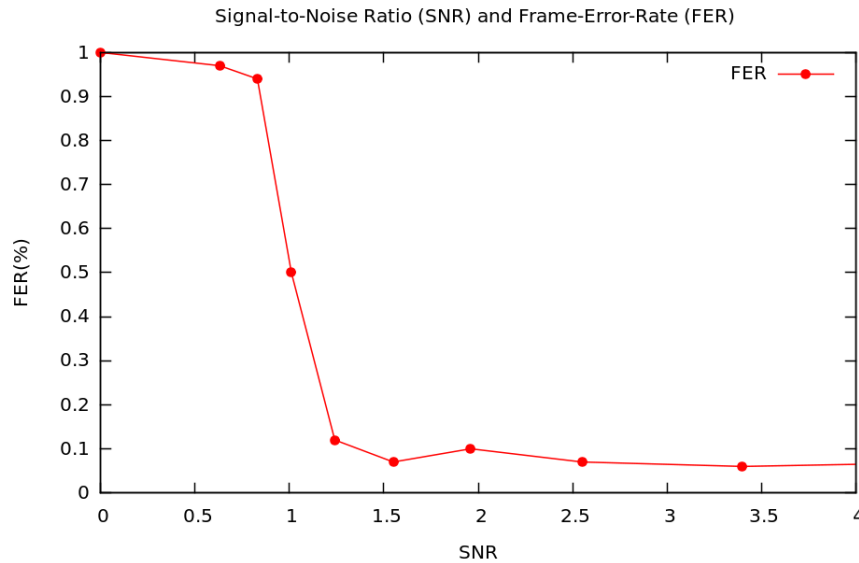


FIGURE 4.1 – FER en fonction de SNR

#### 4.1.2 Algorithme 2 : Anticipation basée sur la prédiction de la puissance du signal reçu

Nous présentons les différentes mesures et données requises pour notre approche avant de présenter l'algorithme.

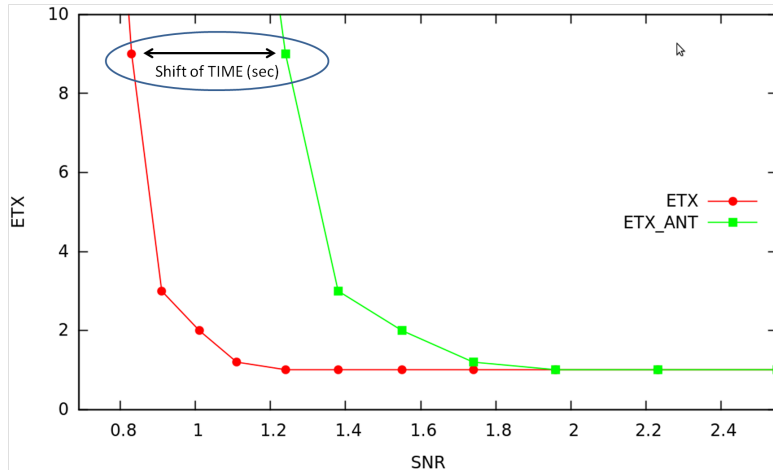


FIGURE 4.2 – ETX et ETX\_ANT en fonction de SNR

#### 4.1.2.1 Relations entre FER et la puissance du signal

L'anticipation se fait par un mécanisme de prédiction sur la base de l'historique des mesures de qualité de liaison. Les métriques ETX et ETT sont principalement basées sur le taux d'erreur de trame (FER) de la liaison sans fil. Par conséquent, c'est cette quantité que nous essayons d'anticiper. Mais, nous n'utilisons pas de mesures précédentes de FER car il se comporte comme une fonction en escalier (step function). Par contre, la puissance du signal reçu est strictement décroissante/croissante car les nœuds se déplacent et elle est directement liée au FER. Il est donc plus facile de prévoir une tendance pour la puissance du signal que pour FER.

Nous collectons et conservons la puissance du signal des paquets échangés entre les voisins pour chaque lien. Ces mesures sont stockées dans un tableau et sont utilisées pour calculer la métrique du lien. Cette quantité est généralement disponible par l'intermédiaire du Received Signal Strength Indicator (RSSI) dans la norme 802.11 et les cartes 3G/LTE. Pour des raisons d'implémentation, ces mesures ne sont collectées que pour des paquets de contrôle. En effet, le démon de routage peut ne pas être informé de la réception de paquets de données. Les futures puissances du signal sont estimées à partir de ces mesures. Comme précédemment nous supposons que chaque nœud dispose d'une table de mapping reliant la puissance du signal et le FER. Ce tableau peut être obtenu par le fabricant de la carte réseau ou déduit par un ensemble de mesures effectuées avant le déploiement du réseau ad hoc. Une fonction notée  $FER(.)$  est alors utilisée pour le mapping de la puissance du signal et le FER. Nous supposons également que les prédictions de FER sont échangées dans les paquets HELLO du protocole de routage. Elle permettra au nœud de prédire le FER dans les deux directions d'une liaison.



#### 4.1.2.2 Anticipation de la métrique

L'idée de base de l'algorithme est similaire à celui décrit en section 4.1.1. Il consiste à utiliser les métriques classiques ETX ou ETT lorsque la qualité de la liaison est bonne, et à augmenter artificiellement et progressivement les valeurs des métriques lorsque les liens commencent à se détériorer.

L'algorithme est présenté dans l'algorithme 4. Il considère la métrique ETX, mais on pourrait aussi envisager ETT. La nouvelle mesure est notée *ETX\_ANT*. L'algorithme est appliqué à chaque réception d'un paquet de contrôle. À chaque réception, le récepteur mesure *LINK\_Q*.

---

**Algorithm 4: Management Mobility ALGORITHM**


---

```
void ETXcalculation()
begin
    /* The different variables are updated at each HELLO reception */
    Update LINK_Q with the last Hello measurement;

    if LINK_Q > TH_Q then
        /* The link quality is good or is improving: we do not anticipate */
        ETX_ANT = COMPUTE_ETX();
    else
        /* The quality is under the TH_Q threshold: we anticipate */
        SIGNAL_PRED = SIGNAL_PREDICTION();
        FER_PRED = FER(SIGNAL_PRED);
        ETX_ANT = 1/(FER_PRED*d_f);
```

---

La variable *LINK\_Q* est comparée à un seuil prédéfini noté *TH\_Q*. Nous anticipons la valeur d'ETX, seulement si la qualité du lien est inférieure à ce seuil. En premier lieu, la puissance du signal est prédite avec la fonction *SIGNAL\_PREDICTION(.)* à l'aide d'un historique des mesures passées (précédentes) de la puissance du signal. La méthode utilisée pour prédire la puissance du signal est décrite dans la section suivante.

Ensuite, nous faisons correspondre cette future puissance de signal à un FER attendu avec la fonction *FER(.)*. Il permet au nœud d'estimer le paramètre  $d_r$  de la métrique ETX (voir Section 2.4.1).  $d_f$  est calculé sur l'autre extrémité de la liaison (prédit ou non dépendant de la qualité de liaison), et obtenu au moyen du routage des paquets de contrôle. La métrique *ETX\_ANT* est alors calculée à partir de ces valeurs.

#### 4.1.2.3 Algorithme de prédiction de la puissance du signal

Plusieurs méthodes ont été proposées pour prédire la force du signal [112]. La plupart de ces méthodes suppose un environnement radio particulier, par perte de trajet ou par évanouissement donné.

Les performances d'un système sans fil dépendent de l'environnement dans lequel il fonctionne à cause du comportement du canal radio. La méthode de régression linéaire a été proposée [113] dans le but de développer un mécanisme de prédiction non spécifique à un site de qualité de liaison sans fil et de valider ses performances dans des environnements différents. En effet, cette méthode est indépendante de l'environnement de propagation et de la distance entre les nœuds d'utilisateurs et les points d'accès. Elle est à faible complexité, mais avec un mécanisme de prédiction précis pour prévoir la qualité de la liaison sans fil à longue portée.

La méthode de régression linéaire proposée utilise comme entrée les mesures précédentes de la puissance du signal, afin de prédire la future tendance de la puissance du signal reçu. Cette méthode adopte un mécanisme de fenêtrage adaptatif pour s'adapter à des changements brusques dans la trace de données, ce qui réduit considérablement l'erreur de prédiction. Cet algorithme a été testé dans divers environnements, sur des réseaux réalistes, et dans plusieurs endroits différents : intérieurs et extérieurs [113].

Les auteurs de [113] comparent cette méthode à l'un des meilleurs algorithmes de prédiction de canal existants. Ils montrent que cet algorithme peut être utilisé comme un indicateur relativement plus précis et robuste. En résumé, nous avons choisi cette méthode de prédiction du signal pour trois raisons : faible complexité, pas d'hypothèses sur les canaux radio, et des tests ont montré qu'elle était efficace. Dans la suite, nous considérons cette méthode, mais notre approche de métrique anticipée peut très bien considérer d'autres méthodes de prédiction du signal.

#### 4.1.2.4 Méthode de régression linéaire

Avec une méthode de régression linéaire, un historique des mesures de signal, noté  $S = \{s_1, s_2, \dots, s_n\}$ , est stocké dans une table représentant les dernières  $n$  mesures mises à jour à chaque nouvelle réception. Les différents moments au cours desquels les mesures sont collectées, sont stockées dans le vecteur  $T = \{t_1, t_2, \dots, t_n\}$ . Un modèle de régression linéaire de la forme :  $\bar{S} = a + bT$  est utilisé pour ajuster les données. La future puissance du signal estimée est alors comme suit :

$$\hat{s}_{n+p} = a + bp \quad (4.1.3)$$

Où  $p$  est le pas de la prediction.  $a$  et  $b$  sont donnés par :

$$a = \bar{s} - b\bar{t} \quad (4.1.4)$$

$$b = \frac{\sum_{st} - \bar{s} \cdot \bar{t}}{\sum_t^2 - n\bar{t}^2} \quad (4.1.5)$$

$\sum st$  est la covariance de  $S$  et  $T$ .  $\bar{s}$  (respectivement  $\bar{t}$ ) est la moyenne du vecteur  $S$  (respectivement  $T$ ).  $\sum_t^2$  est le moment d'ordre 2 :  $\sum_t^2 - n\bar{t}^2$  est la variance de  $T$ .

L'algorithme 5 décrit la méthode de prédiction, comme défini dans [113]. En raison du mouvement aléatoire du nœud lorsqu'il se déplace et l'accompagnement des changements brusques, les auteurs de [113] notent que la trace de la puissance du signal reçue ne peut pas être facilement traçable. Pour cette raison, ils utilisent une fenêtre dynamique (taille dynamique des vecteurs  $S$  et  $T$ ) pour s'adapter aux changements brusques. Lors de ces changements brusques, les "anciennes" données ne reflètent pas la tendance actuelle de l'environnement. La réduction de la taille de la fenêtre est donc nécessaire afin d'écarter les mesures qui ne sont plus pertinentes. L'utilisation d'une fenêtre glissante adaptée aux variations brusques de la puissance du signal est le grand avantage de cet algorithme. La taille de la fenêtre d'historique est initialement réglée à une valeur par défaut, et les variations selon les erreurs sont observées dans le processus de prédiction. Si l'erreur de prédiction est supérieure à un seuil, la taille de la fenêtre est immédiatement réduite. Cependant, si l'erreur ne dépasse pas le seuil donné, la taille de la fenêtre est augmentée jusqu'à ce qu'elle atteigne sa taille maximale.

---

**Algorithm 5: SIGNAL\_PREDICTION ALGORITHM** [113]

---

```
void SIGNAL_PREDICTION()
```

```
begin
```

```

    p : prediction step;
     $\hat{y}_{i+p}$  : prediction of  $(i + p)_{th}$  sample
    w : window size
    wstart : starting window size;
    wmax : maximum window size;
    e : prediction error;
    INITIALIZATION;
    w = wstart ;
    /* i is a loop control index */
    i = w;
    PREDICTION;
    while 1 do
        /* read new incoming measurement */
         $\hat{y}_{i+p} = a + bp$ ;
         $e = |\hat{y}_i - y_i|$ ;

        if e > emax then
            w = wstart;
        else
            if w < wmax then
                w ++;
            else
                w = wmax;
            i ++;
    end while
end
```

---

#### 4.1.2.4.1 Expérimentations

Nous avons évalué l'algorithme de prédiction décrit dans la section précédente par une série d'expérimentations. Nous avons configuré deux ordinateurs portables en mode ad hoc sous Linux (Ubuntu 12.04 LTS). Ils étaient équipés de deux cartes USB sans fil externes. Nous avons effectué cette évaluation dans le jardin extérieur de notre laboratoire. Les deux ordinateurs portables ont été mis dans le champ de vision l'un de l'autre. Premièrement, les deux ordinateurs sont déplacés dans des directions opposées à une vitesse constante de piétons, puis ils sont restés plus ou moins à la même distance avant de les approcher quelques secondes et de les éloigner après pour une deuxième fois. Cette expérimentation vise à évaluer l'algorithme de prédiction quand il y a des mouvements clairs dans la puissance du signal (nœuds s'approchant ou nœuds s'éloignant) et lorsque le signal oscille (la distance entre les deux PC ne varie pas beaucoup).

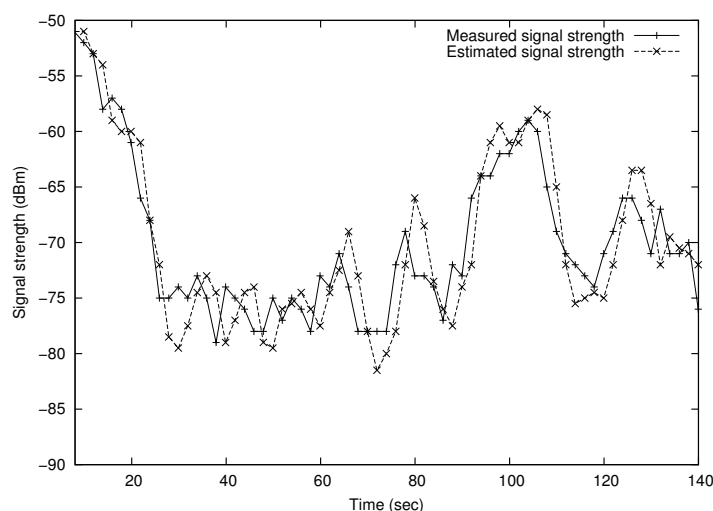


FIGURE 4.3 – Comparaison entre la puissance du signal réelle mesurée et la puissance du signal prédite (2 secondes à l'avance)

Dans la figure 4.3, nous avons tracé la puissance du signal mesurée et la puissance du signal estimée (2 secondes à l'avance). La courbe représentant le signal anticipé est recalé sur celle mesurée : à un instant  $t$  les deux courbes représentent le signal mesuré à ce moment et la valeur que l'on avait anticipée il y a 2 secondes. Elle montre que ce mécanisme réussit à prédire la puissance du signal lorsque les nœuds s'approchent et s'éloignent. Comme prévu, lorsque la puissance du signal oscille, la prédiction est moins précise. Plusieurs expériences similaires ont été réalisées et ont conduit au même comportement.

#### 4.1.2.5 Configuration des paramètres

Avant de procéder à l'évaluation de notre proposition, nous devons régler les paramètres ANTICIPATION\_TIME et  $TH_Q$ . On note ANTICIPATION\_TIME (exprimé en secondes) la variable

qui correspond au temps de prédiction (le nombre de secondes que nous anticipons la puissance du signal/métrique).  $ANTICIPATION\_TIME$  est lié à la variable  $p$  dans l'équation 4.1.3. Il doit être réglé en fonction du temps nécessaire pour la diffusion des nouvelles valeurs des métriques dans le réseau et le temps de traitement (calcul de route, etc.) Par conséquent, ce paramètre dépend fortement du protocole de routage utilisé. Dans nos simulations, nous avons mis  $ANTICIPATION\_TIME$  à 2 secondes d'une part 1 seconde est insuffisante pour anticiper et d'autre part la courbe tracée précédemment dans la section 4.1.2.4.1 confirme ce choix.

L'autre paramètre,  $TH_Q$ , doit être réglé de façon à ce que la prédiction commence au moins  $ANTICIPATION\_TIME$  secondes avant une rupture de lien. Par conséquent, un réglage fin de ce paramètre prend en compte la connaissance de l'environnement radio et la vitesse relative maximale des nœuds mobiles. Dans nos scénarios, cette information est connue. Dans le cas d'un déploiement réel, ils peuvent être préalablement déduits par expérimentations.

Dans l'hypothèse où le seuil de la réception physique est atteint au moins  $ANTICIPATION\_TIME$  secondes après le seuil  $LINK\_Q$ , nous obtenons :

$$TH_{Rec} \geq PL(PL^{-1}(TH_Q) + S_{rel} \times ANTICIPATION\_TIME) \quad (4.1.6)$$

Où  $PL(distance)$  est la fonction d'atténuation (path-loss) qui estime la puissance du signal reçu en fonction de la distance,  $TH_{Rec}$  est le seuil physique pour la réception de la trame, et  $S_{rel}$  est la vitesse relative entre deux nœuds. Nous obtenons :

$$TH_Q \geq PL(PL^{-1}(TH_{Rec}) - S_{rel} \times ANTICIPATION\_TIME) \quad (4.1.7)$$

## 4.2 Évaluation des Performances

L'algorithme 1 décrit dans 4.1.1 et l'algorithme 2 décrit dans 4.1.2 servent à anticiper les métriques ETX et ETT. Nous comparons ces deux algorithmes en effectuant des simulations approfondies avec le simulateur NS-3. Nous considérons trois scénarios (décrits précédemment dans la section 3.3 du chapitre 3) : une chaîne de nœuds, un réseau maillé et un réseau totalement mobile ad hoc.

Dans tous les scénarios, nous utilisons le modèle de perte de trajet Log Distance [114] comme modèle de propagation radio. La fonction d'atténuation (donnée en dBm) est :

$$PL(d) = \begin{cases} P_t - 10\alpha \log_{10}(d/d_0) - c & \text{when } d \geq d_0 \\ P_t - c & \text{when } d \leq d_0 \end{cases} \quad (4.2.8)$$

Parameters	Numerical Values
Simulation time	62 [sec]
Window size (Chain scenario)	$1200 \times 200 [m^2]$
Window size (Mesh and Ad hoc scenario)	$300 \times 300 [m^2]$
Number of nodes (Chain scenario)	12
Number of nodes (Mesh scenario)	15
Number of nodes (Ad hoc scenario)	25
Propagation model	Log Distance Propagation Loss Model
Fading	$\sigma^2 = 0, 4, 25$ $n = 0$
Wireless technology	IEEE 802.11a
Traffic type	Constant Bit Rate (CBR)
Packet size	1024 bytes
WIFI_MANAGER	Minstrel_Wifi_Manager
$TH_Q$ for Anticipation with signal strenght prediction	2.23 mWatt
$TH_Q$ for Anticipation as Taylor expansion	1.11 mWatt
HELLO_INTERVAL (OLSR)	0.25 sec
TC_INTERVAL (OLSR)	2 sec
ANTICIPATION_TIME	2 sec

TABLE 4.1 – Simulation Parameters

Où  $d$  est la distance entre l'émetteur et le récepteur,  $P_t$  est la puissance d'émission (en dBm),  $c$  est une constante ( $c = 46, 67$ ),  $\alpha$  présente la décroissance de la puissance en fonction de la distance ( $\alpha = 3$ ), et  $d_0$  est la distance de référence ( $d_0 = 1$  mètre). Cette fonction de perte de trajet est favorable à notre algorithme qui utilise la régression linéaire pour prédire la puissance du signal. Il fonctionne très bien avec un tel modèle déterministe. Par conséquent, nous combinons cette fonction avec une variable  $S$  modélisant l'évanouissement aléatoire et le caractère aléatoire de l'évanouissement du canal.  $S$  suit une distribution log-normale avec une moyenne nulle et différents écarts standards  $\sigma^2$  (varianht de 0 à 25). L'intensité du signal reçu est alors  $PL(d) + S$  (exprimée en dBm).

Les paramètres fixes utilisés pour les simulations sont donnés dans le tableau 4.1. Les paramètres de simulation pour les deux algorithmes sont les mêmes, sauf le paramètre seuil noté  $TH_Q$ . Il prend la valeur 1.11 mWatt pour le premier algorithme basé sur le développement de Taylor, par contre dans le deuxième algorithme il prend la valeur 2.23 mWatt. Le premier algorithme (4.1.1) était notre première proposition, nous avons pris la valeur de  $TH_Q = 1.11$  car le FER augmente fortement quand SNR est inférieure à cette valeur donnée comme nous l'avons expliqué dans la Section 4.1.1. Le deuxième algorithme (4.1.2) était notre deuxième proposition, nous avons choisi la valeur de  $TH_Q = 2.23$  dans ce cas, pour donner une marge de temps suffisante à l'anticipation de la métrique.

Pour chaque simulation, nous avons pris une seule source et une seule destination qui sont choisies au hasard parmi les nœuds mobiles. Chaque point sur les courbes est la moyenne de 20 simulations et il est représenté avec un intervalle de confiance de 95%. Les positions initiales

de la source et la destination sont différentes d'une simulation à l'autre ainsi que leur trajectoire puisqu'elles sont tirées aléatoirement.

### 4.2.1 Topologie chaîne

Nous mesurons le PDR, la capacité et le délai de bout-en-bout des différentes métriques pour la topologie chaîne. Nous détaillons les résultats obtenus dans le paragraphe suivant.

#### 4.2.1.1 PDR

est le rapport entre le nombre de paquets reçus par la destination et le nombre de paquets envoyés par la source.

Dans les scénarios de simulation, la source envoie deux paquets par seconde à la destination. Nous mesurons le PDR du protocole OLSR pour chaque test. Nous avons opté pour un trafic très léger afin d'éviter les problèmes de congestion qui mènent à des pertes de messages de contrôle et des routes obsolètes.

La figure 4.4 montre le PDR en utilisant différentes métriques avec l'algorithme 1 en variant la vitesse du nœud mobile sans évanouissement du signal ( $\sigma^2 = 0$ ). Les performances des métriques ETX\_ANT et ETT\_ANT surpassent les autres métriques. Pour des vitesses plus élevées, les performances d'ETX\_ANT et ETT\_ANT diminuent, parce que l'algorithme 1 a été dimensionné pour une vitesse maximale de 50km/h. LD est la pire métrique car elle sélectionne les plus longs chemins en terme de durée.

La figure 4.5 montre le PDR en utilisant différentes métriques avec l'algorithme 2. Le PDR avec ETX\_ANT est à 100% pour toutes les vitesses. Cela prouve de façon empirique que la mobilité des nœuds est parfaitement gérée pour ce scénario. Pour une vitesse plus élevée, les performances des ETT\_ANT diminuent, mais elles sont supérieures à 98,5% et surpassent les métriques classiques. Les performances inférieures d'ETT\_ANT sont dues au Wi-Fi (paramètre  $B$  dans l'équation 2.4.5) qui n'est pas anticipé dans notre algorithme, et peut conduire à des prévisions inexactes. LD sélectionne les chemins avec la plus longue durée cela explique pourquoi elle est la pire métrique.

Pour illustrer l'impact du fading, nous avons tracé le PDR en fonction de la vitesse du nœud mobile des deux métriques ETX\_ANT et ETX pour l'algorithme 1 (voir figure 4.6) et l'algorithme 2 (voir figure 4.7). Nous observons que les performances se dégradent lorsque la variance augmente quelle que soit la métrique pour les deux algorithmes. Mais, les performances d'ETX\_ANT sont meilleures que les performances d'ETX quel que soit l'algorithme d'anticipation utilisé. En effet, avec  $\sigma^2 = 4$ , le PDR avec ETX chute jusqu'à 16% par contre le PDR

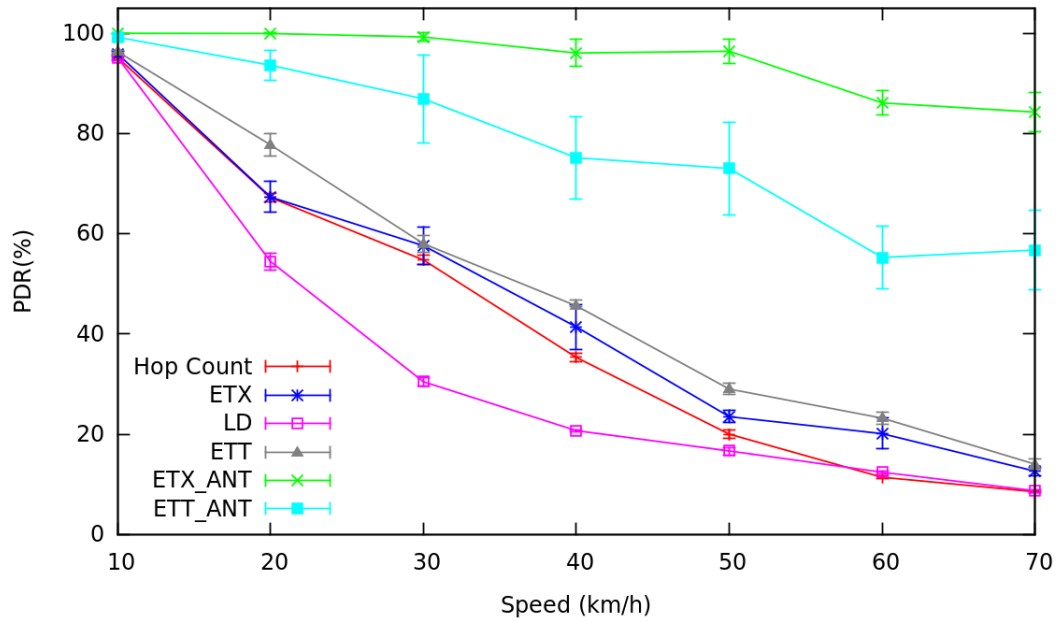


FIGURE 4.4 – PDR en fonction de la vitesse avec l'algorithme 1 (Chaîne de noeuds)

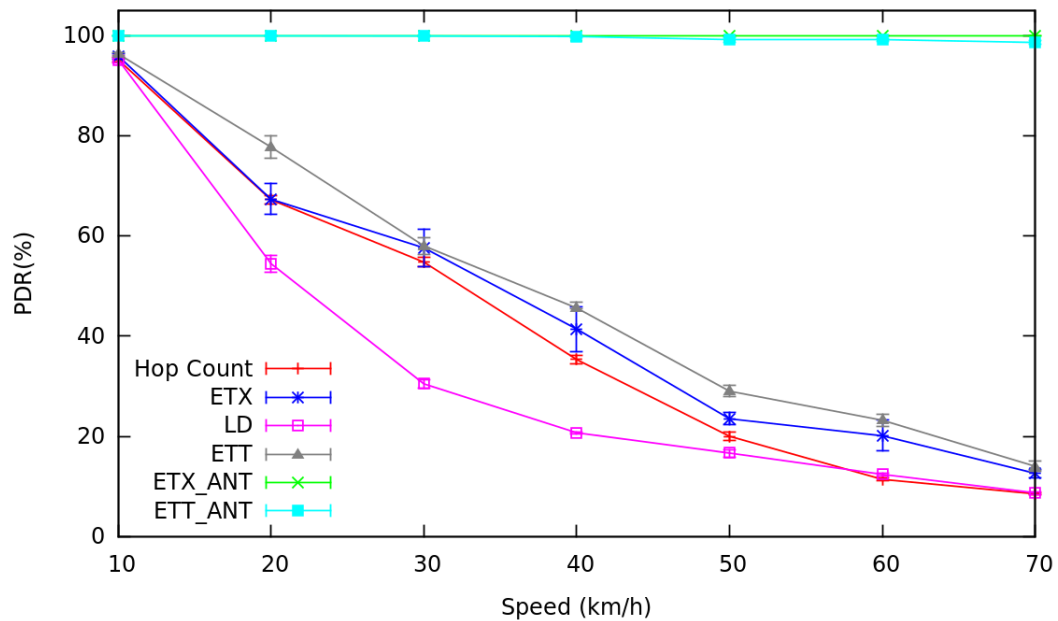


FIGURE 4.5 – PDR en fonction de la vitesse avec l'algorithme 2 (Chaîne de noeuds)

avec ETX\_ANT dépassent 93% et 38% respectivement pour l'algorithme 2 et l'algorithme 1. Et avec  $\sigma^2 = 25$ , le PDR avec ETX\_ANT dépassent 24% avec les deux algorithmes d'anticipation, par contre, le PDR avec ETX atteint 20%.

Nous observons dans la figure 4.8 que les performances d'ETX\_ANT avec l'algorithme 2 sont meilleures en terme de PDR qu'ETX\_ANT avec l'algorithme 1. En effet, ETX\_ANT est toujours proche d'un PDR de 100% quand  $\sigma^2 = 4$  avec l'algorithme 2. Avec un fading faible ( $\sigma^2 = 4$ ), PDR avec ETX\_ANT reste supérieure à 93% montrant que la mobilité est bien gérée. Pour  $\sigma^2 =$



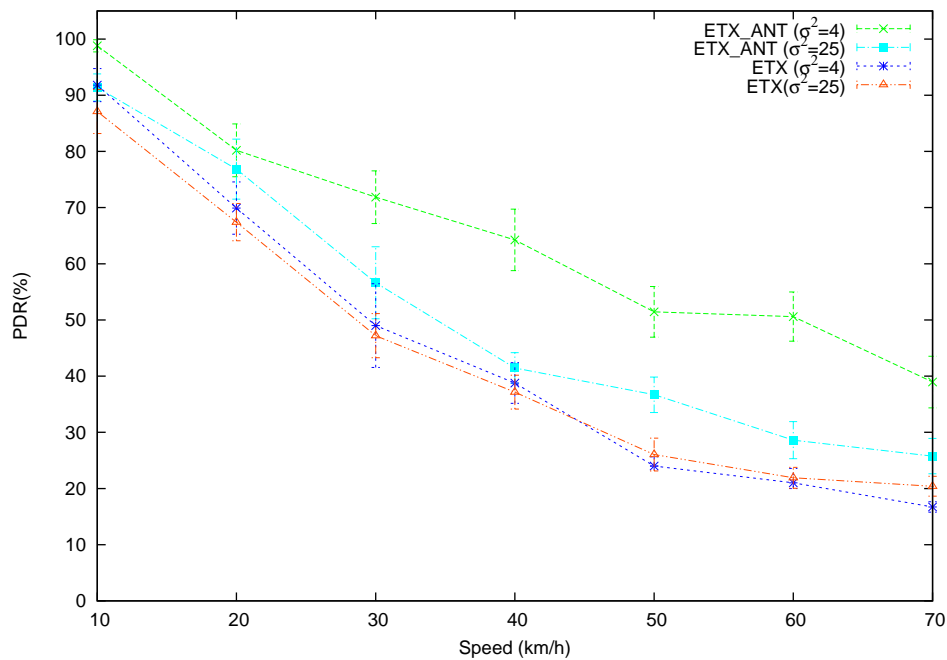


FIGURE 4.6 – Algorithme 1 : PDR en fonction de la vitesse avec du fading (Chaîne de noeuds)

25, le PDR est d'au plus 24%. Environ 75% des pertes de paquets sont dues à l'évanouissement et non à la mobilité (c'est le pourcentage que nous avons observé lorsque les nœuds n'étaient pas mobiles). Cela correspond à un scénario pire cas où le fading est très important, très aléatoire, et donc par nature imprévisible.

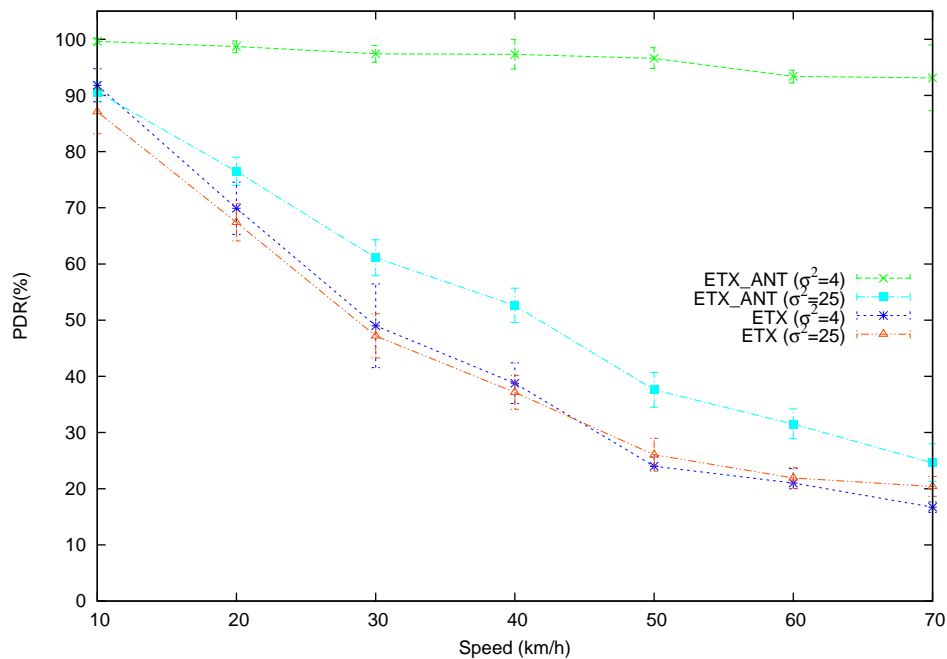


FIGURE 4.7 – Algorithme 2 : PDR en fonction de la vitesse avec du fading (Chaîne de noeuds)

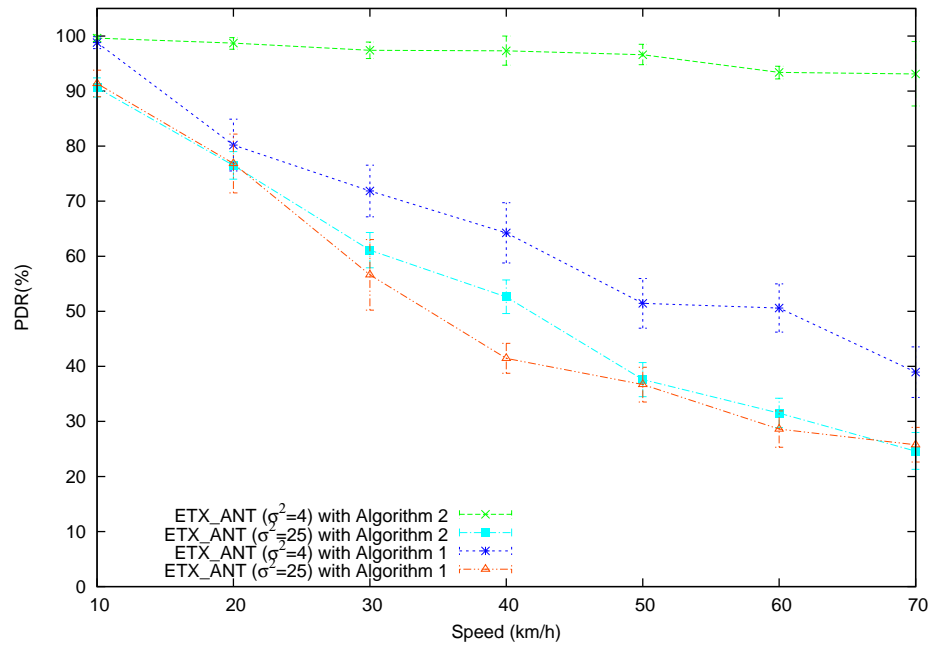


FIGURE 4.8 – Comparaison des deux algorithmes : PDR en fonction de la vitesse du nœud mobile avec du fading (Chaîne de noeuds)

#### 4.2.1.2 Capacité

elle reflète la charge de trafic écoulé d'un réseau : il s'agit de la moyenne du débit en réception pour un couple (source, destination). Nous avons mesuré la capacité obtenue pour les deux algorithmes d'anticipation à une vitesse fixe de  $25\text{km/h}$ , afin de les comparer pour la topologie chaîne.

Les mesures de performances des métriques anticipées en terme de capacité avec l'algorithme 1 surpassent celles des métriques classiques. Nous observons dans la figure 4.9 que le débit de toutes les métriques atteint  $680\text{Kb/s}$ .

La figure 4.10 montre le débit en fonction du débit de la source pour l'algorithme 2. Les mesures de performance des métriques ETX\_ANT et ETT\_ANT sont meilleures que les mesures de performances pour les autres métriques. Cela prouve que la technique d'anticipation augmente la capacité du réseau afin d'assurer une qualité de service meilleure pour les utilisateurs.

Afin de comparer la capacité des métriques anticipées pour les deux algorithmes, nous avons tracé dans la figure 4.11 le débit en fonction du débit de la source. Nous remarquons que les performances d'ETX\_ANT et d'ETT\_ANT avec l'algorithme 2 sont légèrement meilleures que les performances d'ETX\_ANT et d'ETT\_ANT avec l'algorithme 1. En revanche, les métriques anticipées atteignent le débit de  $680\text{Kb/s}$  quel que soit l'algorithme d'anticipation utilisé.

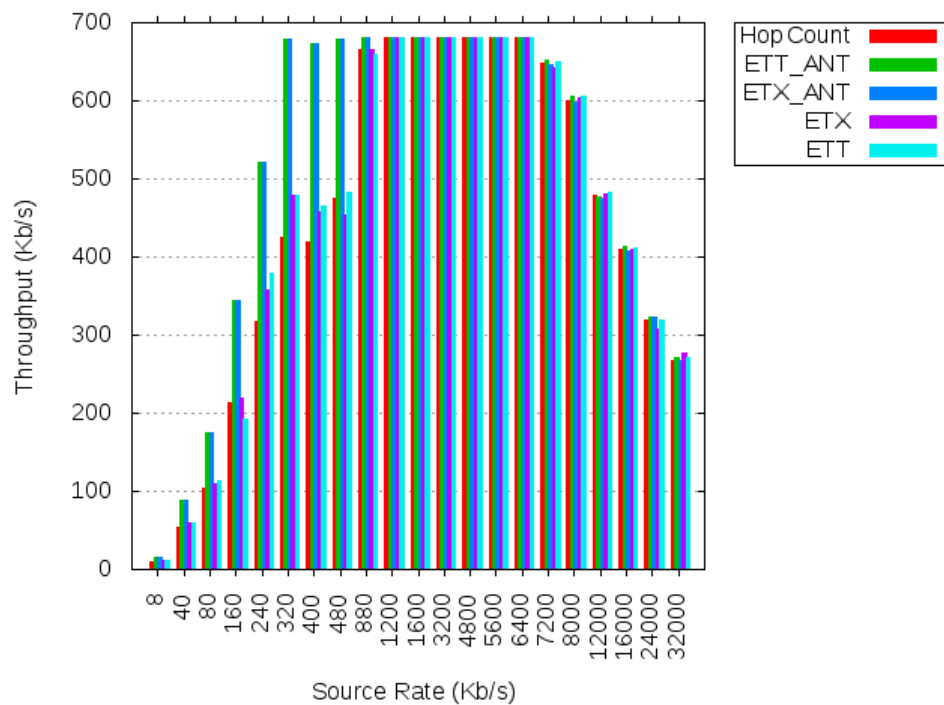


FIGURE 4.9 – Algorithme 1 : capacité de toutes les métriques (Chaîne de noeuds)

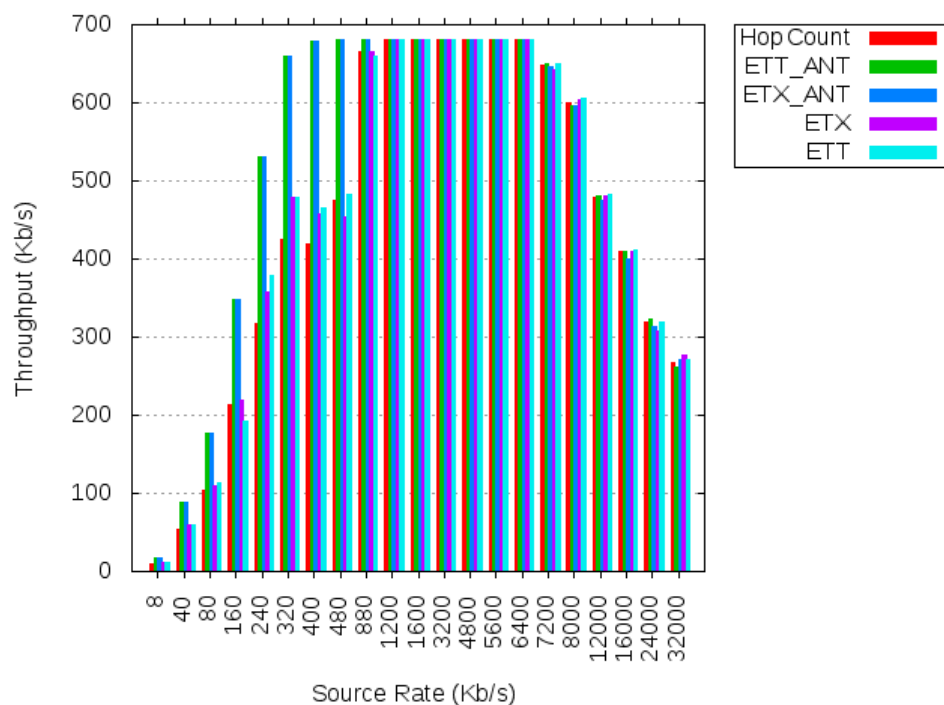
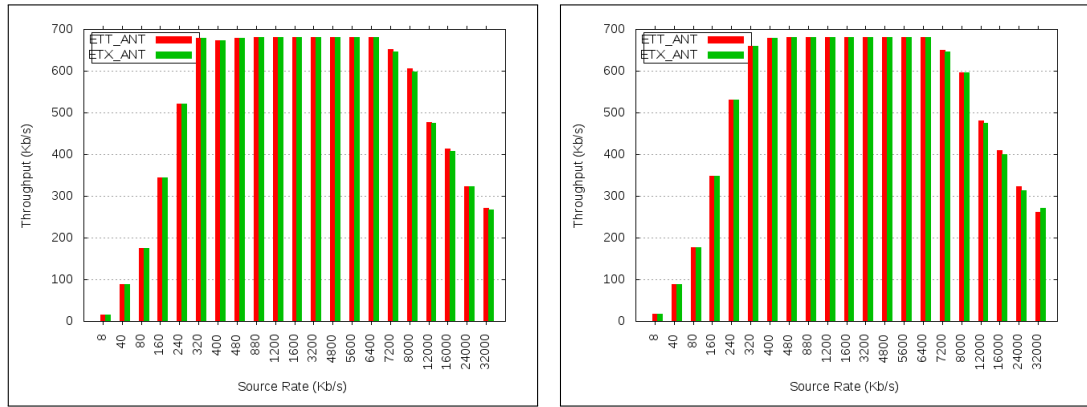


FIGURE 4.10 – Algorithme 2 : capacité de toutes les métriques (Chaîne de noeuds)

#### 4.2.1.3 Délai de bout-en-bout

Considérons maintenant, la mesure du délai de bout-en-bout qui reflète le délai entre l'émission du paquet par la source et sa réception par la destination. Nous avons effectué des simulations



(a) Algorithme 1 : capacité des métriques anticipées ETX\_ANT (b) Algorithme 2 : capacité des métriques anticipées ETX\_ANT et ETT\_ANT.

FIGURE 4.11 – Capacité des métriques anticipées (Chaîne de noeuds)

pour les métriques Hop Count, ETX, ETT, LD et les métriques anticipées avec l'algorithme 1 et l'algorithme 2.

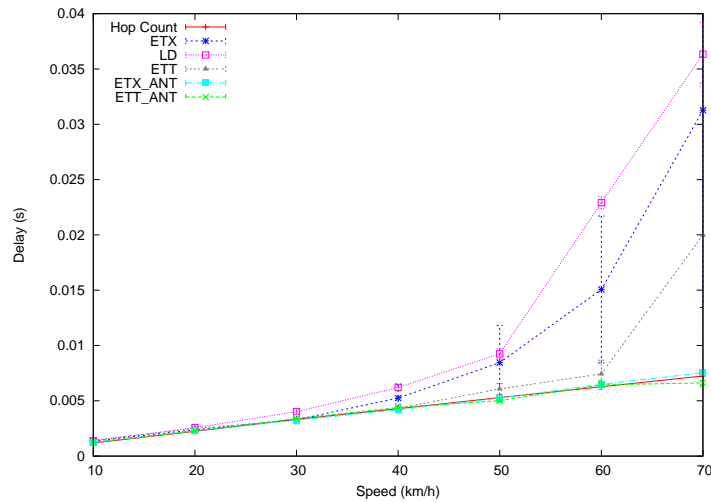


FIGURE 4.12 – Algorithme 1 : Délai de bout-en-bout pour toutes les métriques (Chaîne de noeuds)

La figure 4.12 montre le délai exprimé en secondes en fonction de la vitesse des nœuds pour l'algorithme 1. Comme nous l'avons expliqué (dans la section 3.3 du chapitre 3), le délai de Hop Count est moins important que le délai d'ETX et d'ETT car ces dernières choisissent des routes plus longues. Les métriques ETX\_ANT et ETT\_ANT distribuent les paquets sur différents chemins plus sûrs (sans perte ou avec les meilleurs coûts), ce qui peut effectivement réduire le délai. En effet, nous avons vu que le PDR chutait pour les autres métriques, amenant des retransmissions et donc un délai supplémentaire.

Pour l'algorithme 2, le délai de bout-en-bout est illustré dans la figure 4.13. Les métriques classiques génèrent des délais plus importants que celles d'ETX\_ANT et d'ETT\_ANT. Cela

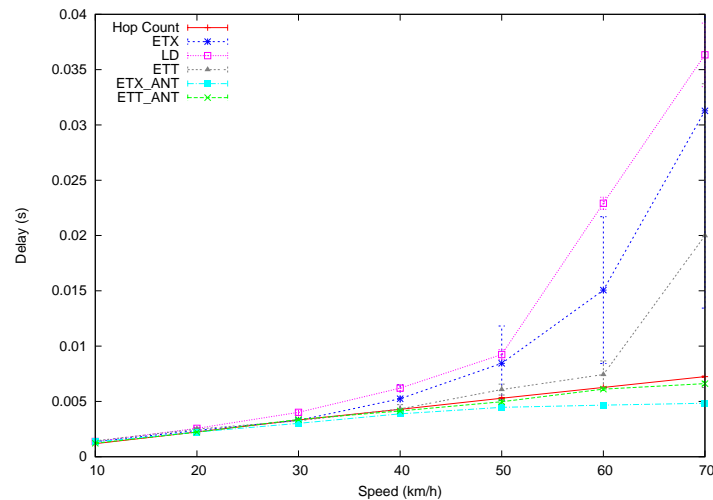


FIGURE 4.13 – Algorithme 2 : Délai de bout-en-bout pour toutes les métriques (Chaîne de noeuds)

se traduit par une sélection de routes plus solides et avec moins de pertes pour ETX\_ANT et ETT\_ANT par rapport aux autres métriques.

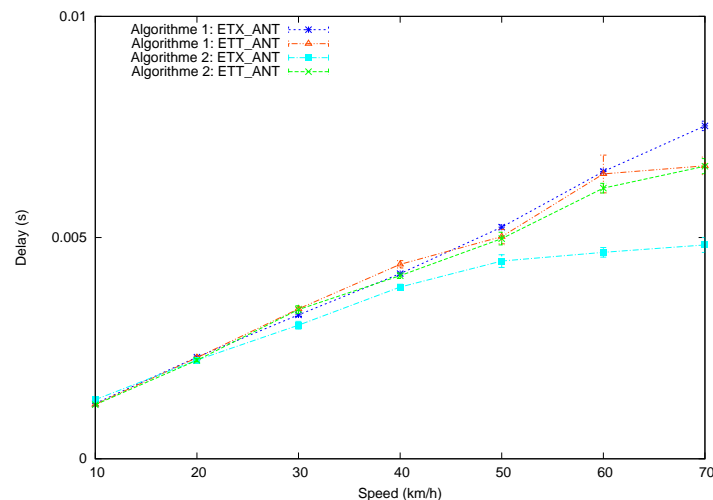


FIGURE 4.14 – Délai de bout-en-bout pour les métriques anticipées (Chaîne de noeuds)

La figure 4.14 montre une comparaison entre les métriques anticipées respectivement pour l'algorithme 1 et l'algorithme 2. La première constatation est que les métriques anticipées réussissent à avoir un délai minimum par rapport aux autres métriques quel que soit l'algorithme d'anticipation utilisé. Le délai avec ses métriques ne dépasse pas 0.0075s. Cela signifie que les métriques anticipées favorisent les chemins les plus robustes et les plus stables. La deuxième constatation est que les délais d'ETX\_ANT et d'ETT\_ANT avec l'algorithme 2 sont légèrement meilleurs que ceux avec l'algorithme 1. Cela peut s'expliquer par une meilleure gestion de la mobilité de l'algorithme 2 par rapport à l'algorithme 1.

### 4.2.2 Topologie mesh

Dans ce paragraphe, nous évaluons les performances des différentes métriques par rapport à la topologie mesh pour les deux algorithmes d'anticipation. Nous utilisons les mêmes métriques de performances pour la topologie chaîne : le PDR, la capacité et le délai de bout-en-bout.

#### 4.2.2.1 PDR

Les résultats quantitatifs sont similaires au cas chaîne. Les métriques ETX\_ANT et ETT\_ANT montrent des PDR proches de 100% en présence de mobilité (figure 4.15 et figure 4.16) et sont nettement meilleurs que les métriques classiques. Par rapport au résultat de la topologie chaîne, nous observons que le PDR est meilleur pour l'algorithme 1 pour nos 2 métriques. Nous expliquons ceci par une meilleure couverture du réseau mesh.

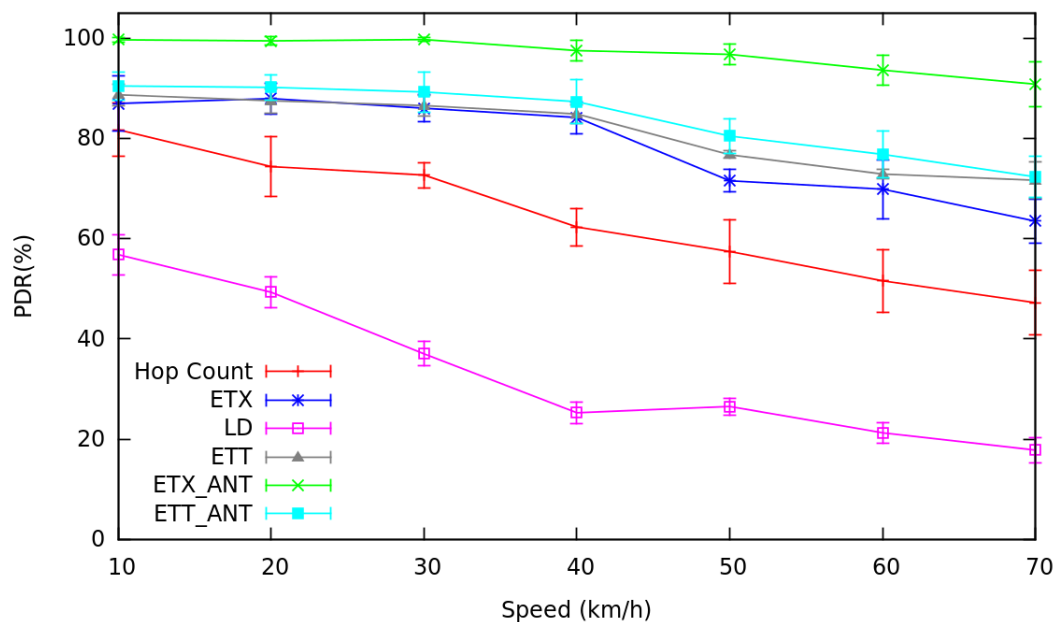


FIGURE 4.15 – PDR en fonction de la vitesse avec l'algorithme 1 (Mesh)

La figure 4.17 et la figure 4.18 montrent les performances des deux métriques ETX\_ANT et ETX en présence du fading avec l'algorithme 1 et l'algorithme 2.

La figure 4.19 compare les deux algorithmes. Il apparaît clairement que là aussi l'algorithme 2 montre de meilleures performances. Les PDR sont aussi plus élevés que pour la topologie chaîne due à une meilleure couverture offerte par le réseau mesh.

Pour montrer la différence de routage entre ETX et ETX\_ANT, le tableau 4.2 illustre la table de routage du noeud 10.1.0.15 avec les deux métriques pour la topologie mesh. Dans ce scénario, le noeud 10.1.0.10 est la source et le noeud 10.1.0.15 est la destination.

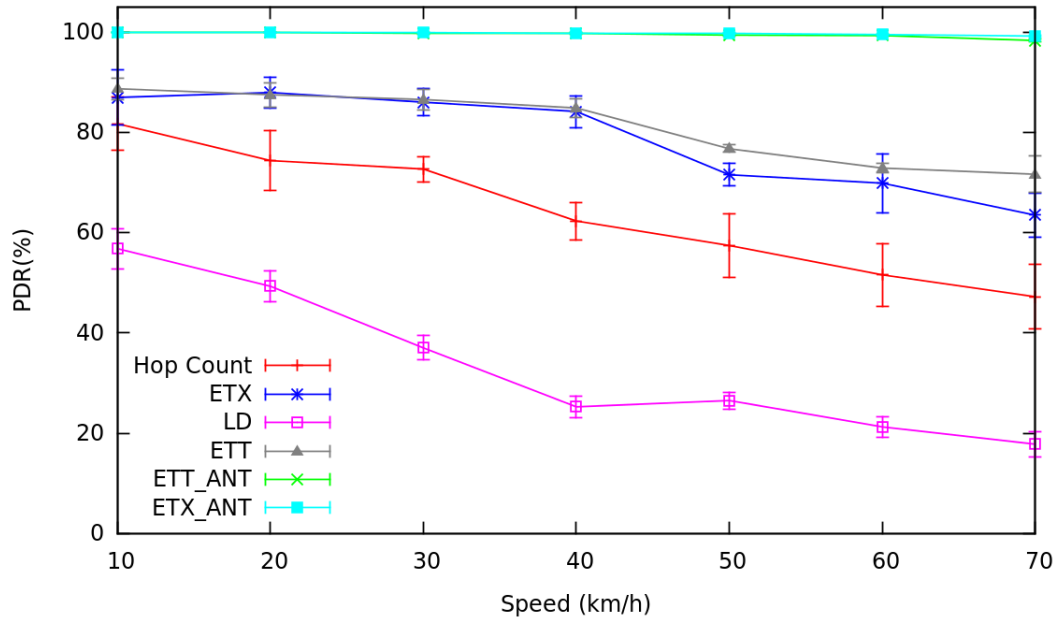


FIGURE 4.16 – PDR en fonction de la vitesse avec l'algorithme 2 (Mesh)

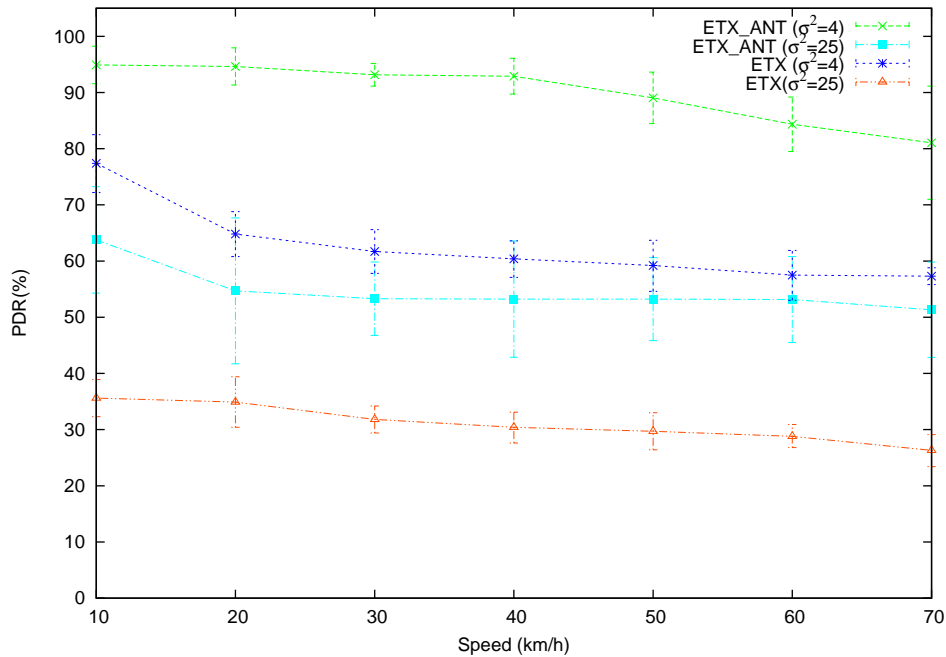


FIGURE 4.17 – Algorithme 1 : PDR en fonction de la vitesse des nœuds mobiles avec du fading (Mesh)

Nous avons affiché dans le tableau 4.2 la table de routage du nœud 10.1.0.15 à la même position des nœuds avec les deux métriques. Nous observons dans le tableau, qu'il existe des valeurs du champ "Metric" qui sont les mêmes pour ETX et ETX\_ANT tel que les nœuds 10.1.0.3, 10.1.0.6 et 10.1.0.9. Ce qui prouve l'utilisation d'ETX\_ANT de la métrique ETX lorsque le SNR est élevé. Par contre, il existe plusieurs valeurs du champ "Metric" d'ETX\_ANT qui sont différentes de celle d'ETX, tel que 10.1.0.1, 10.1.0.2, etc. Par exemple pour le nœud

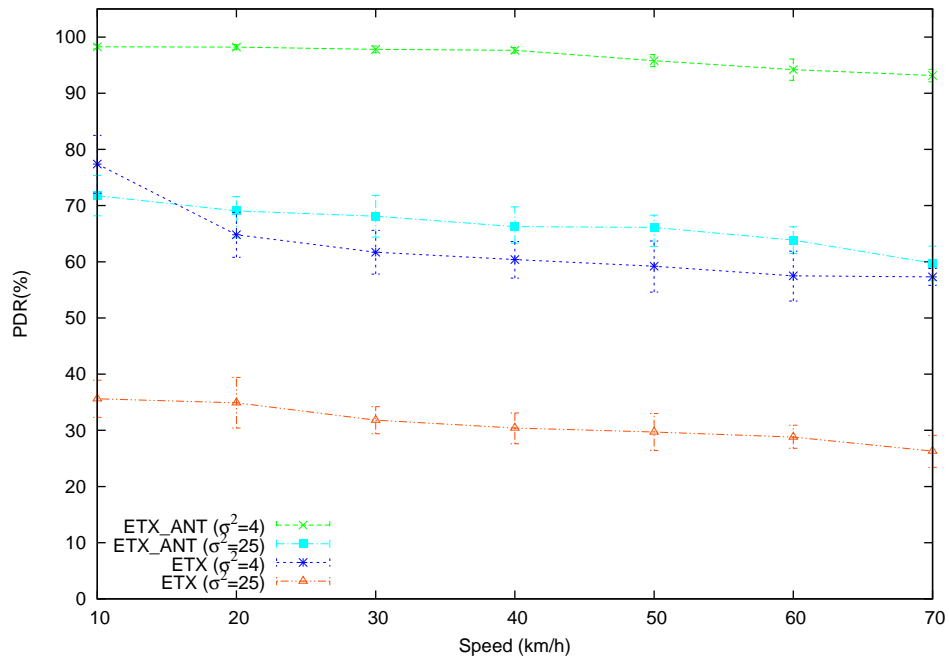


FIGURE 4.18 – Algorithme 2 : PDR en fonction de la vitesse des nœuds mobiles avec du fading (Mesh)

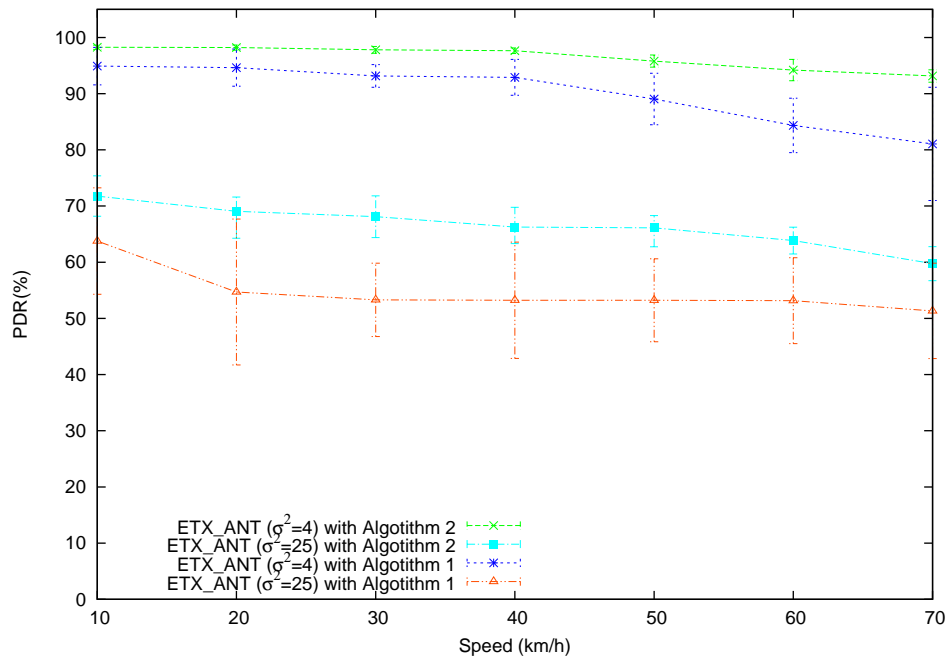


FIGURE 4.19 – Comparaison des deux algorithmes : PDR en fonction de la vitesse des nœuds mobiles avec du fading (Mesh)

10.1.0.1, ETX et ETX\_ANT proposent deux routes différentes. La route sélectionnée par ETX est 10.1.0.15  $\rightarrow$  10.1.0.2  $\rightarrow$  10.1.0.1 avec un coût de 2,25. Par contre, la route choisie par ETX\_ANT est 10.1.0.15  $\rightarrow$  10.1.0.3  $\rightarrow$  10.1.0.1 avec un coût de 3,5. Ce changement de route est dû à la différence de métrique. Nous remarquons que les valeurs de la métrique ETX\_ANT



ETX			ETX_ANT		
Node	Metric	Next Hop	Node	Metric	Next Hop
10.1.0.1	2,25	10.1.0.2	10.1.0.1	3,5	10.1.0.3
10.1.0.2	1,25	10.1.0.2	10.1.0.2	2,5	10.1.0.3
10.1.0.3	1,5	10.1.0.3	10.1.0.3	1,5	10.1.0.3
10.1.0.4	2,25	10.1.0.2	10.1.0.4	3	10.1.0.6
10.1.0.5	2,25	10.1.0.2	10.1.0.5	2	10.1.0.6
10.1.0.6	1	10.1.0.6	10.1.0.6	1	10.1.0.6
10.1.0.7	3	10.1.0.2	10.1.0.7	4	10.1.0.6
10.1.0.8	2	10.1.0.6	10.1.0.8	3	10.1.0.6
10.1.0.9	2	10.1.0.6	10.1.0.9	2	10.1.0.6
10.1.0.10	2,25	10.1.0.2	10.1.0.10	3,5	10.1.0.3
10.1.0.11	2	10.1.0.6	10.1.0.11	3	10.1.0.6
10.1.0.12	3	10.1.0.14	10.1.0.12	4	10.1.0.6
10.1.0.13	2	10.1.0.14	10.1.0.13	3	10.1.0.6
10.1.0.14	1	10.1.0.14	10.1.0.14	2,5	10.1.0.3

TABLE 4.2 – Table computing pour le nœud destination 10.1.0.15 à  $T = 2061$  millisecondes avec ETX et ETX\_ANT

sont pondérées par rapport aux valeurs de la métrique ETX. Ce qui prouve que l'algorithme d'anticipation augmente la valeur de la métrique ETX lorsque le SNR est faible.

#### 4.2.2.2 Capacité

Nous avons tracé aussi la capacité du réseau mesh pour toutes les métriques avec les deux algorithmes, sans fading.

Nous observons le débit en fonction du débit de la source, avec l'algorithme 1, dans la figure 4.20 et dans la figure 4.21 pour l'algorithme 2.

La figure 4.22 compare la capacité des deux algorithmes. Il est clair que là aussi les performances de l'algorithme 2 sont meilleures en terme de débit.

#### 4.2.2.3 Délai de bout-en-bout

Les figures 4.23 et 4.24 montrent le délai de bout-en-bout pour les différentes métriques. Comme pour la topologie chaîne, le délai reste stable pour nos deux métriques anticipées alors qu'il explose pour les autres métriques quand la mobilité devient forte. Les délais sont légèrement différents pour nos deux algorithmes d'anticipation (voir figure 4.25).

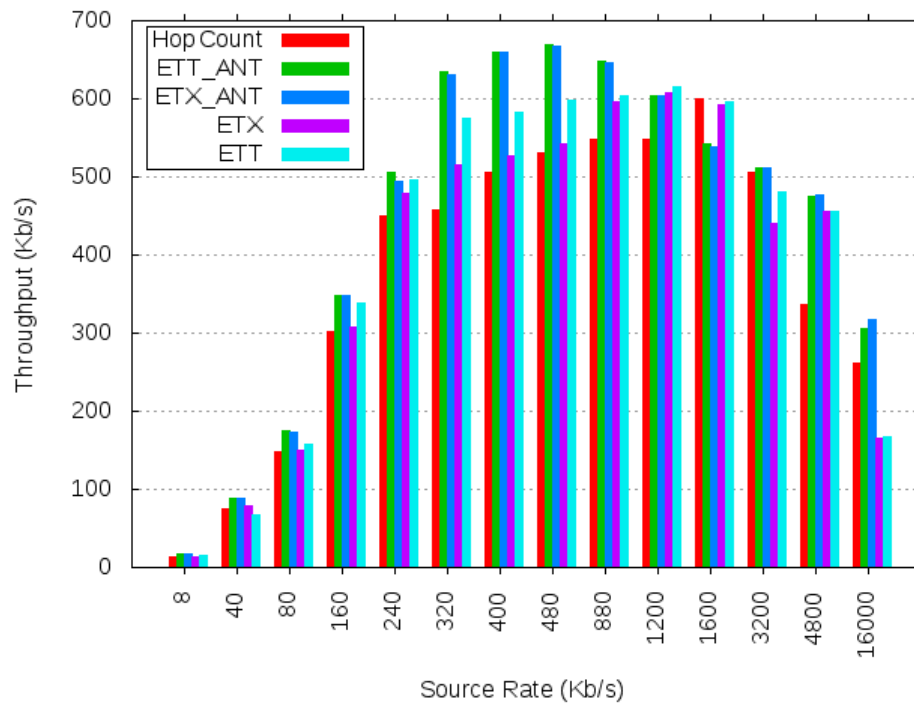


FIGURE 4.20 – Algorithme 1 : capacité de toutes les métriques (Mesh)

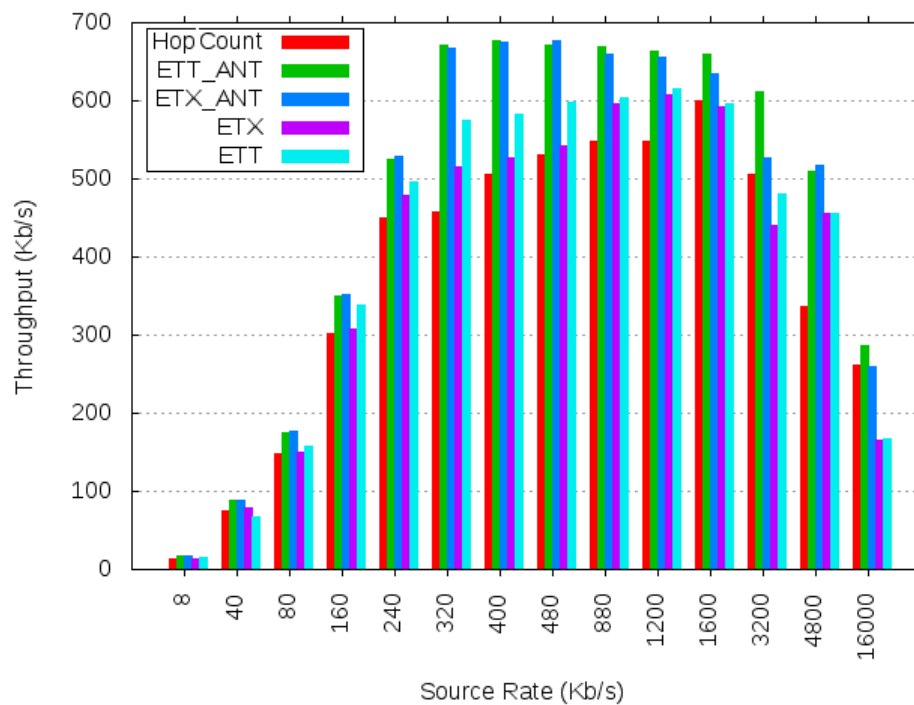


FIGURE 4.21 – Algorithme 2 : capacité de toutes les métriques (Mesh)

### 4.2.3 Topologie aléatoire

Enfin, nous évaluons aussi les performances du réseau mobile ad hoc pour l'ensemble des métriques avec les deux algorithmes en terme de : PDR, capacité et délai de bout-en-bout.

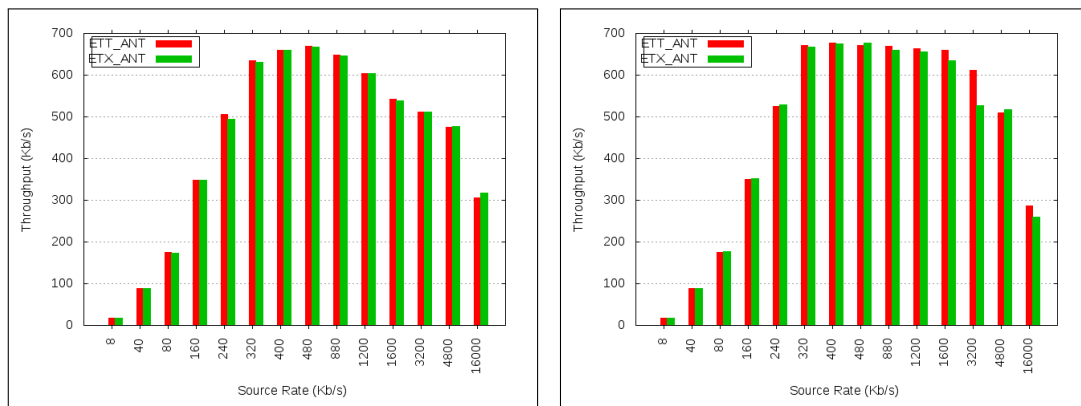


FIGURE 4.22 – Capacité des métriques anticipées (Mesh)

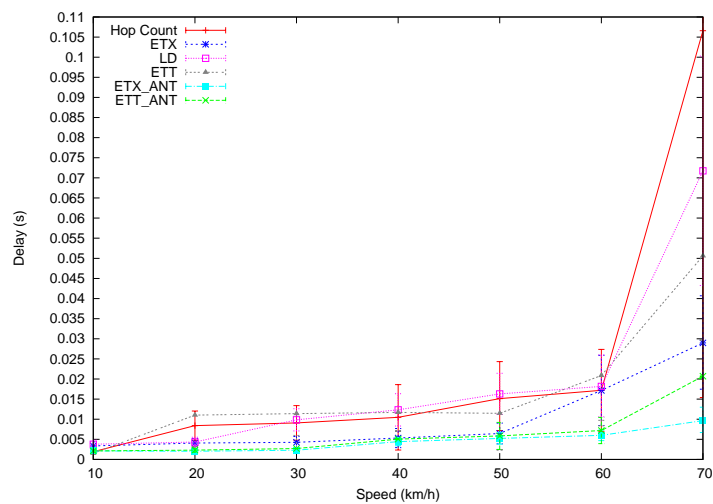


FIGURE 4.23 – Algorithme 1 : Délai de bout-en-bout pour toutes les métriques (Mesh)

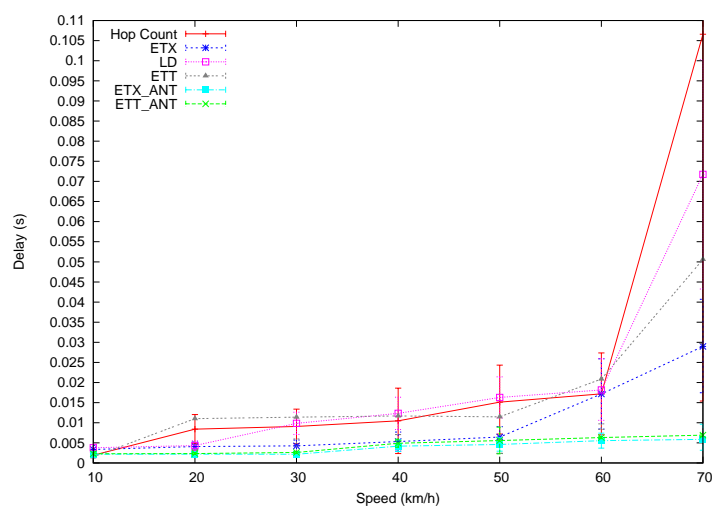


FIGURE 4.24 – Algorithme 2 : Délai de bout-en-bout pour toutes les métriques (Mesh)

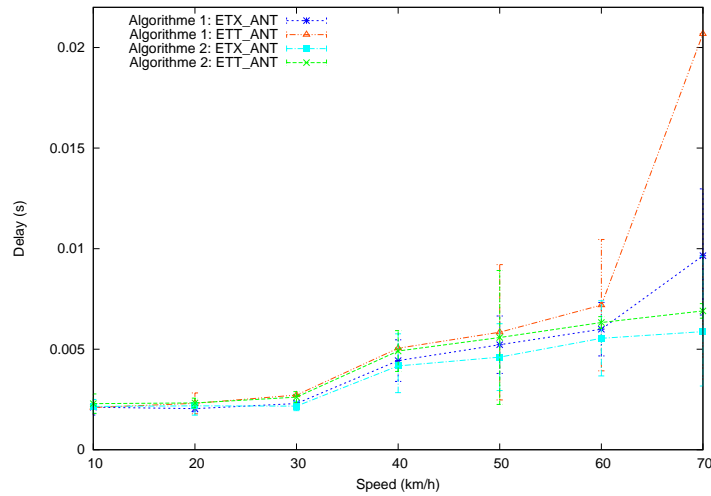


FIGURE 4.25 – Délai de bout-en-bout des métriques anticipées (Mesh)

#### 4.2.3.1 PDR

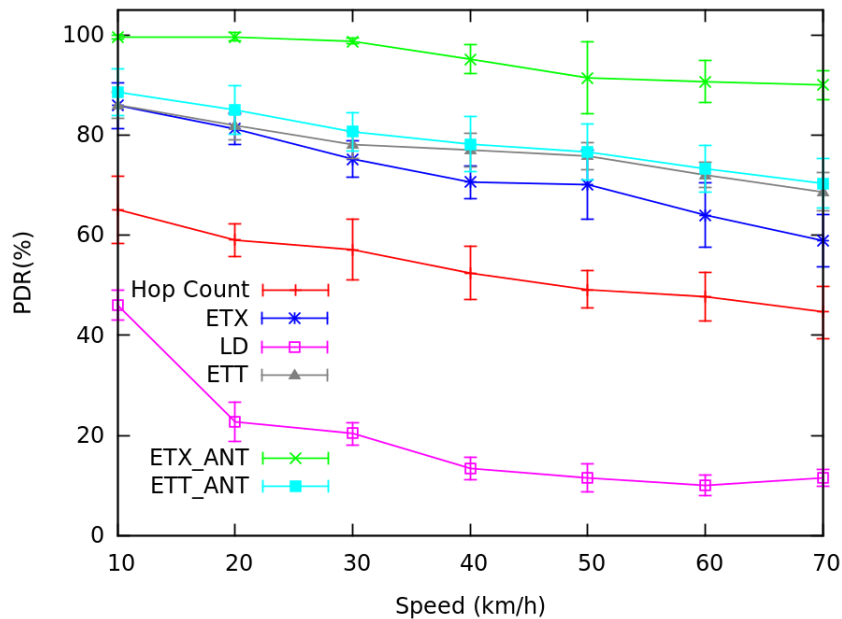


FIGURE 4.26 – PDR en fonction de la vitesse avec l'algorithme 1 (Topologie aléatoire)

Dans la figure 4.27, nous observons que la métrique ETX anticipée est toujours très proche de 100%. Le PDR avec ETX\_ANT ne chute pas en dessous de 97% et ETT\_ANT ne dépasse pas 92%. La métrique LD présente les pires performances : le PDR chute jusqu'à 11,5%.

Afin d'observer l'impact du fading pour la topologie ad hoc, nous avons présenté le PDR en fonction de la vitesse des nœuds mobiles pour les deux métriques ETX\_ANT et ETX (voir figure 4.28 et figure 4.29). Les résultats sont équivalents aux autres topologies. Pour  $\sigma^2 = 4$ , le PDR reste supérieur à 90% pour l'algorithme 2 et 85% pour l'algorithme 1, quelle que soit la vitesse.

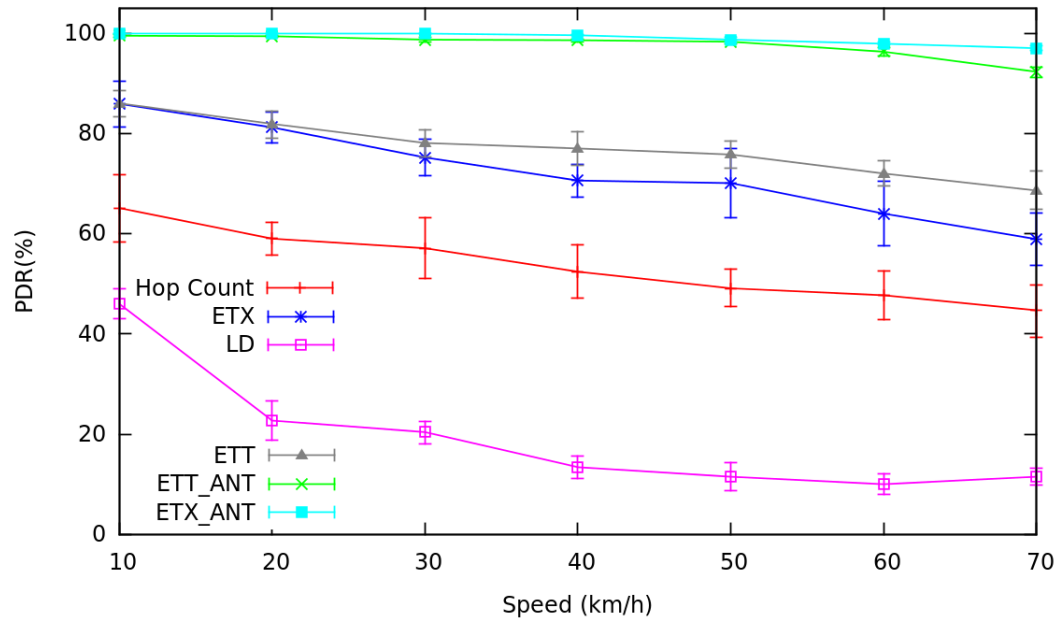


FIGURE 4.27 – PDR en fonction de la vitesse avec l’algorithme 2 (Topologie aléatoire)

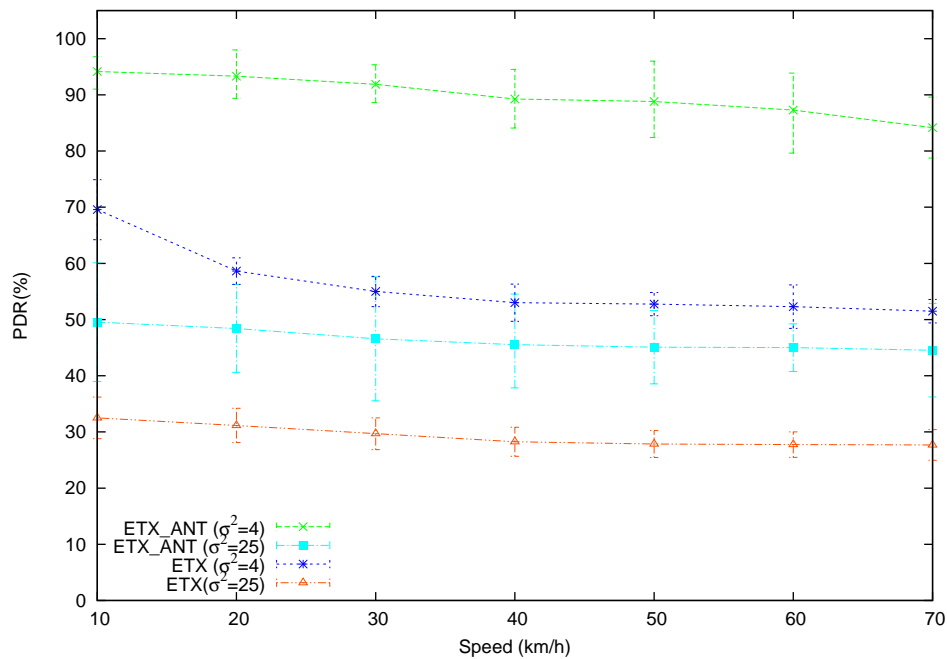


FIGURE 4.28 – Algorithme 1 : PDR en fonction de la vitesse des nœuds mobiles avec du fading (Topologie aléatoire)

Pour un fading supérieur et pour les métriques ETX et ETT classiques, les performances sont médiocres. La figure 4.30 compare les deux algorithmes pour les deux mêmes valeurs de fading. Il apparaît clairement que l’algorithme 2 offre de meilleures performances. Quand le fading est important ( $\sigma^2 = 25$ ), les pertes sont liés à ce fading et non plus à la mobilité : pour ETX\_ANT (algorithme 2), le PDR ne décroît que de 50% entre une vitesse de 10 et 70km/h.

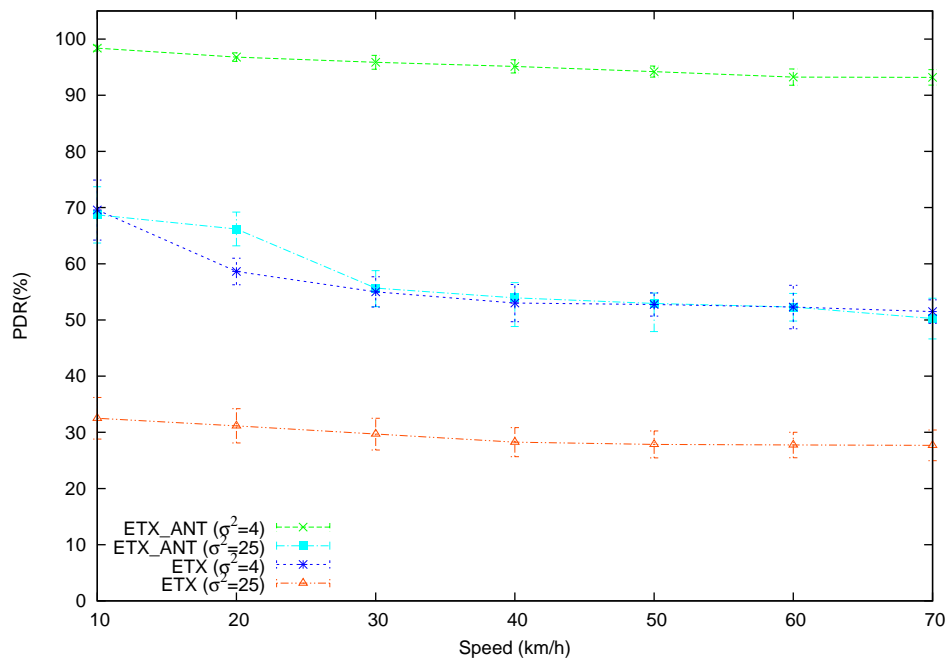


FIGURE 4.29 – Algorithme 2 : PDR en fonction de la vitesse des nœuds mobiles avec du fading (Topologie aléatoire)

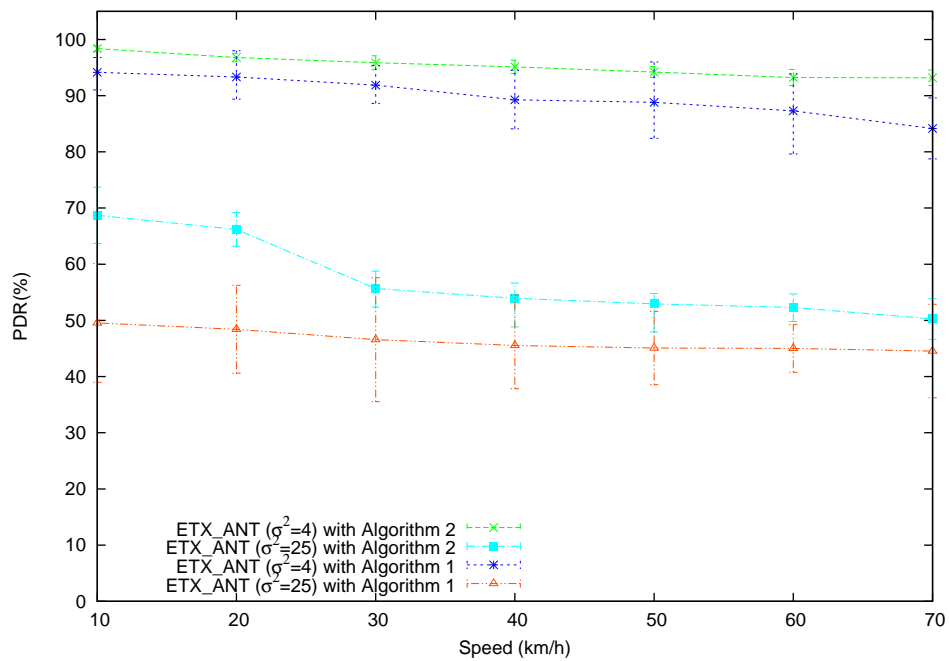
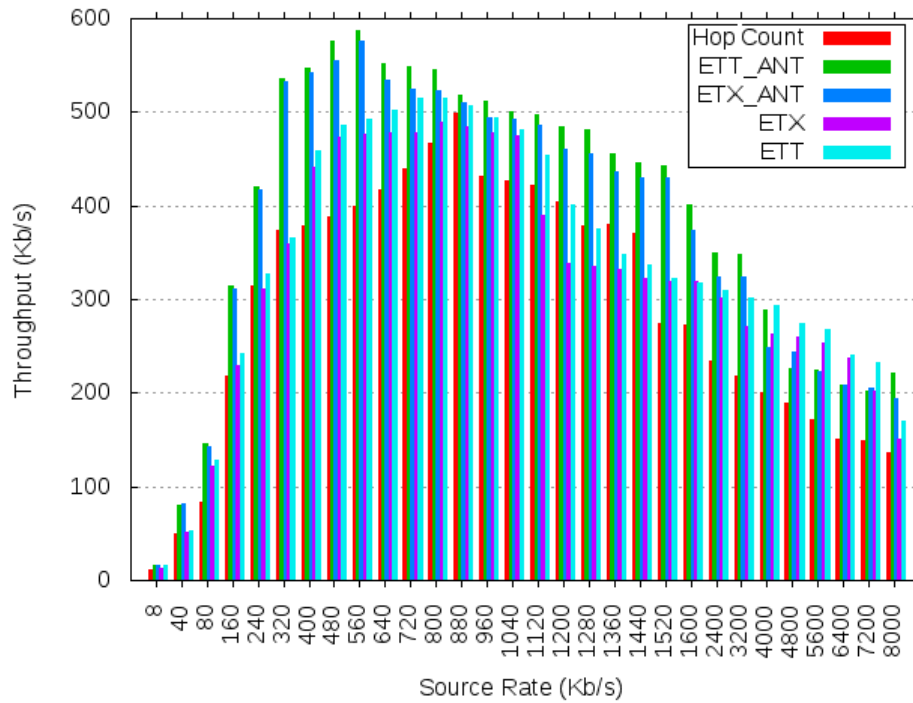
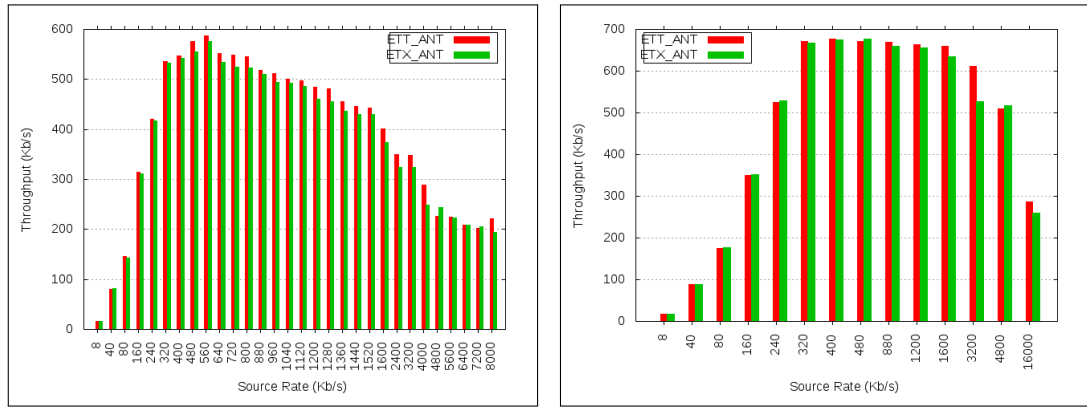


FIGURE 4.30 – Comparaison des deux algorithmes : PDR en fonction de la vitesse des nœuds mobiles avec du fading (Topologie aléatoire)

#### 4.2.3.2 Capacité

Pour ce scénario, nous traçons aussi la capacité avec une vitesse égale à  $25\text{km/h}$  pour les nœuds mobiles.





(a) Algorithme 1 : capacité des métriques anticipées ETX\_ANT et ETT\_ANT. (b) Algorithme 2 : capacité des métriques anticipées ETX\_ANT et ETT\_ANT.

FIGURE 4.33 – Capacité des métriques anticipées (Topologie aléatoire)

La figure 4.33 montre la capacité des métriques anticipées pour les deux algorithmes. Dans le cas de l'algorithme 1, le débit des métriques anticipées atteint  $586\text{Kb/s}$ . En revanche, dans le cas de l'algorithme 2, le débit des métriques anticipées dépasse  $600\text{Kb/s}$ .

#### 4.2.3.3 Délai de bout-en-bout

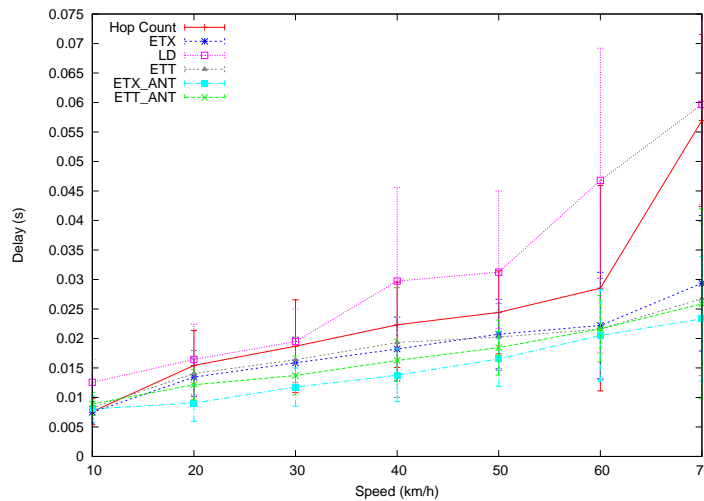


FIGURE 4.34 – Algorithme 1 : Délai de bout-en-bout pour toutes les métriques (Topologie aléatoire)

Les figures 4.34 et 4.35 s'intéressent à l'évaluation du délai en fonction de la vitesse des nœuds mobiles pour les deux algorithmes.

La figure 4.36 est une comparaison des métriques ETX\_ANT et ETT\_ANT pour l'algorithme 1 et l'algorithme 2 en terme de délai de bout-en-bout. Nous observons que les métriques anticipées pour l'algorithme 2 présentent des meilleures délais par rapport à l'algorithme 1.



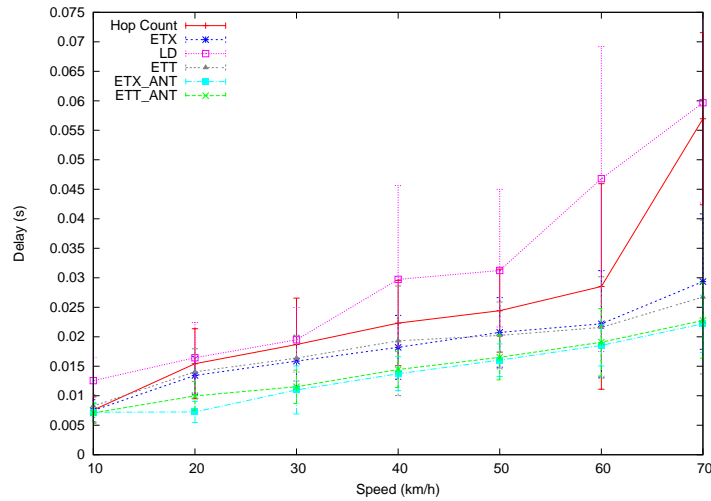


FIGURE 4.35 – Algorithme 2 : Délai de bout-en-bout pour toutes les métriques (Topologie aléatoire)

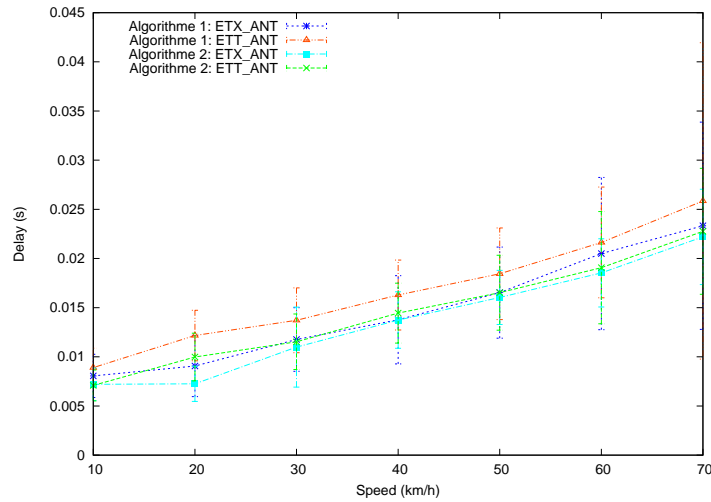


FIGURE 4.36 – Délai de bout-en-bout des métriques anticipées (Topologie aléatoire)

Dans cette section, nous avons présenté une évaluation des performances des métriques classiques et anticipées pour les deux algorithmes. Nous avons simulé trois types de topologie selon le PDR (avec et sans fading), la capacité et le délai de bout-en-bout. Les résultats de simulation montrent que les performances des métriques anticipées surpassent celles des métriques classiques.

### 4.3 Conclusion

Au cours de ce chapitre, nous avons décrit deux algorithmes d'anticipation des métriques de routage. Chacun d'entre eux est basé sur le calcul des métriques ETX et ETT. L'idée de base des deux algorithmes est d'utiliser le calcul classique d'ETX (ou le calcul classique d'ETT)

lorsque la qualité du lien est bonne, et d'augmenter artificiellement et progressivement la métrique lorsque la qualité du lien se dégrade. Un mapping reliant la qualité du lien mesurée à la couche physique et le FER est utilisé dans les deux algorithmes.

Les deux algorithmes diffèrent dans le système de prédiction : un développement de Taylor pour l'algorithme 1, une régression linéaire pour l'algorithme 2. La validation a été effectuée par un grand nombre de simulations. Trois types de topologie ont été simulées, avec et sans fading et les performances ont été évaluées au travers du PDR, de la capacité et du délai de bout-en-bout. Les principales conclusions sont les suivantes :

- Le routage basé sur ces nouvelles métriques reste stable pour des vitesses allant jusqu'à  $70\text{km/h}$ .
- L'algorithme 2 (régression linéaire) montre systématiquement de meilleures performances que l'algorithme 1 (développement de Taylor).
- Les performances des métriques classiques s'écroulent avec la mobilité.

L'approche considérée est donc efficace et permet de bien gérer la mobilité. Ses performances sont très logiquement liées à l'efficacité de la méthode de prédiction.

## Chapitre 5

# Mise en Oeuvre du TESTBED

### Sommaire

---

<b>5.1</b>	<b>Testbed pour les réseaux ad hoc</b>	<b>123</b>
5.1.1	Testbed	123
<b>5.2</b>	<b>Description de notre testbed</b>	<b>124</b>
5.2.1	Paramètres du testbed	124
5.2.1.1	Configuration de la communication en mode ad hoc	125
5.2.1.2	Installation d'OLSRD	126
5.2.1.3	Implémentation d'ETX_ANT dans OLSRD	127
5.2.1.3.1	Mapping FER < -- > Signal	127
5.2.2	Description du scénario	130
<b>5.3</b>	<b>Résultats et discussion</b>	<b>131</b>
<b>5.4</b>	<b>Conclusion</b>	<b>136</b>

---

Dans le chapitre précédent, nous avons proposé d'anticiper les métriques de routage ETX et ETT dans le but d'adapter les routes en temps réel, en présence de mobilité. Nous avons présenté deux algorithmes de prédiction capables d'anticiper ces métriques. L'algorithme 1 est basé sur le développement de Taylor et l'algorithme 2 est basé sur la régression linéaire. Un grand nombre de simulation a été effectué afin de valider l'efficacité des nouvelles métriques. L'algorithme 2 présente des meilleures performances, il offre des routes plus stables et gère mieux la mobilité. Nous nous proposons de valider l'efficacité de cet algorithme dans un banc d'essai (testbed). Ce chapitre décrit la mise en oeuvre de la métrique ETX\_ANT dans un testbed d'un réseau ad hoc. Il commence par citer quelques travaux relatifs à la validation de métrique de routage dans un testbed. Ensuite, il détaille les paramètres nécessaires pour la mise en place du testbed. Après, il décrit la topologie considérée pour la mesure de performance. Enfin, il discute des résultats obtenus.

## 5.1 Testbed pour les réseaux ad hoc

La simulation est l'approche la plus utilisée pour évaluer les protocoles réseaux. Il est plus rare qu'il soit validé par expérimentation en raison de la difficulté de la mise en oeuvre. Mais cette approche peut induire deux problèmes :

- D'une part, il est difficile de simuler des scénarios réseaux exactement tels qu'un réseau réel, en particulier le modèle de la couche physique. Les auteurs de [115] utilisent une technique avec un coût très élevé basé sur un modèle de traçage de rayons intelligents afin de simuler une couche physique plus réaliste. Une différence dans leurs résultats est constatée entre la couche physique réaliste et le modèle utilisé pour la couche physique (le modèle free space ou le modèle two-ray ground). Aussi, le simulateur NS-3 [91] met en oeuvre un certain nombre de modèles radio mais jamais complètement conforme à la réalité.
- D'autre part, certains paramètres réseau sont plus faciles à réaliser et à paramétrer dans un simulateur qu'en pratique. Par exemple, le délai et la bande passante qui sont des informations supplémentaires utilisées par les protocoles de routage comme métriques de routage [116], sans mentionner comment obtenir ces informations en temps réel. Ce genre d'information peut être facile à extraire à partir du simulateur, mais cette méthode n'est pas très pratique dans le cadre d'une utilisation pratique. Par conséquent, une évaluation du protocole par simulation est intéressante mais pas suffisante. Ainsi, une évaluation par un véritable banc d'essai est nécessaire pour s'assurer de la validation du protocole avec la technique actuelle.

### 5.1.1 Testbed

Dans [117], les auteurs ont mis en place un banc d'essai pour valider leur protocole de routage multi-trajets (**Multiple Path OLSR (MP-OLSR)**) basé sur le protocole OLSR dans le monde réel. Ils ont proposé deux scénarios différents afin de vérifier le protocole MP-OLSR et de le comparer avec OLSR. Les résultats révèlent que MP-OLSR est adaptée aux réseaux mobiles denses avec un trafic important, et pourrait satisfaire les applications multimédia critiques avec des fortes contraintes temporelles.

Une technique pour activer le routage sécurisé dans les MANET est proposée dans [118], en utilisant un "banc d'essai WMN". Les auteurs ont évalué différentes approches cryptographiques pour sécuriser le protocole OLSR, pour fournir une base pour la recherche. Les résultats ont évalué les algorithmes les plus efficaces utilisés pour le chiffrement et l'authentification concernant différents schémas de cryptographie et de hachage.

Dans [119], les auteurs étudient le comportement du protocole OLSR avec différents algorithmes de calcul de la métrique ETX : ETX\_ff (ETX funkfeuer freifunk) et ETX\_float. ETX\_ff

TABLE 5.1 – Cartes Wi-Fi internes utilisées dans le testbed

Types
Intel Corporation Centrino Wireless-N 1000
Qualcomm Atheros AR9285 Wireless Network Adapter
Intel Corporation Centrino Wireless-N 2230

présente l’algorithme de qualité de lien implémenté par défaut dans OLSRD [120]. Dans cet algorithme, le taux de perte de paquets en cours est calculé à l’aide des numéros de séquence des paquets du protocole OLSR. ETX\_float calcule le taux de perte en utilisant un paramètre de vieillissement configurable sur les paquets HELLO. Dans cette étude, les auteurs mènent plusieurs expériences dans un banc d’essai MANET afin de comparer les performances d’ETX\_ff et d’ETX\_float. Ils ont changé la valeur de l’intervalle des messages HELLO et ils étudient l’effet de ses différentes valeurs. Ils mettent en oeuvre deux scénarios expérimentaux : statique et mobile, pour observer le comportement des deux algorithmes pour un nombre différent de sauts dans leur testbed.

La configuration d’un testbed de réseau mesh sans fil fondé sur le protocole de routage OLSRD est décrite dans [121]. Les auteurs mesurent les performances des trois métriques de routage Hop Count, ETX et ETT, dans un testbed indoor formé par des dispositifs MIMO hétérogènes. Les résultats de cette étude montrent que la métrique ETT surpasse ETX et Hop Count d’une manière significative en matière de perte de paquets, de délai de bout en bout, et d’efficacité.

## 5.2 Description de notre testbed

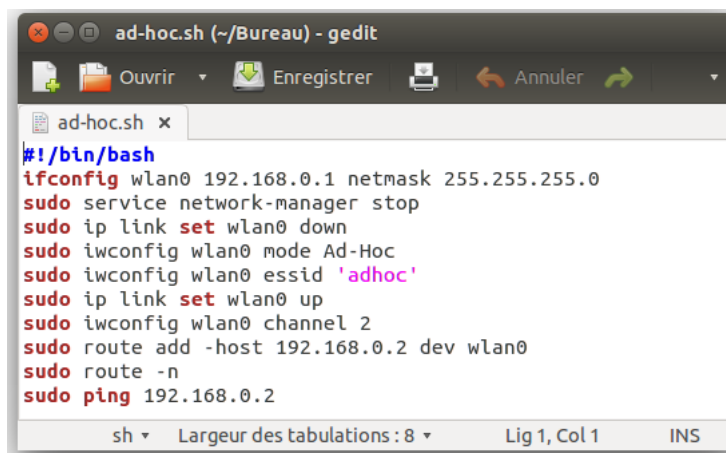
### 5.2.1 Paramètres du testbed

Le testbed est mis en oeuvre dans le but de valider l’efficacité du routage par la métrique ETX\_ANT dans un réseau ad hoc réel. Les logiciels et matériels pour la mise en place du testbed sont résumés ci-dessous :

- Système d’exploitation : Linux Ubuntu 14.04 LTS [122] est installé dans chaque ordinateur portable.
- Interface sans fil : Chaque ordinateur portable possède une carte sans fil pour assurer la communication sans fil en mode ad hoc (IEEE 802.11bgn). Le tableau 5.1 présente les cartes Wi-Fi internes utilisées dans ce testbed.
- Protocole de routage : OLSRD (version 0.6.6.2) [120] est installé dans chaque ordinateur.

### 5.2.1.1 Configuration de la communication en mode ad hoc

Un réseau Wi-Fi ad hoc est constitué uniquement d'un ensemble d'ordinateurs équipés de carte Wi-Fi, sans point d'accès. Les ordinateurs mis en place en mode ad hoc communiquent directement entre eux s'ils sont à portée radio les uns des autres. Sinon, quelques ordinateurs sont choisis comme relais pour assurer la communication entre les ordinateurs hors portée radio. Cette section explique les étapes à suivre pour configurer chaque poste en mode ad hoc. Pour ce faire, il existe deux méthodes, soit en utilisant Network Manager (voir Annexe A.1), soit en exécutant un script Shell. Dans cette section, nous nous sommes intéressé à décrire la configuration d'un réseau en mode ad hoc à partir d'un script Shell.



```
ad-hoc.sh (~/Bureau) - gedit
Ouvrir Enregistrer Annuler
ad-hoc.sh x
#!/bin/bash
ifconfig wlan0 192.168.0.1 netmask 255.255.255.0
sudo service network-manager stop
sudo ip link set wlan0 down
sudo iwconfig wlan0 mode Ad-Hoc
sudo iwconfig wlan0 essid 'adhoc'
sudo ip link set wlan0 up
sudo iwconfig wlan0 channel 2
sudo route add -host 192.168.0.2 dev wlan0
sudo route -n
sudo ping 192.168.0.2
sh Largeur des tabulations : 8 Lig 1, Col 1 INS
```

FIGURE 5.1 – Configuration en mode ad hoc par script Shell

La figure 5.1 montre un exemple de script Shell qui contient un ensemble de commandes qui sont exécutées pour assurer la communication ad hoc. Ces commandes sont :

- **ifconfig wlan0 192.168.0.1 netmask 255.255.255.0** : configure l'interface réseau wlan0 avec l'adresse IP 192.168.0.1 et le masque de réseau 255.255.255.0.
- **service network-manager stop** : arrête le fonctionnement du Network Manager.
- **ip link set wlan0 down** : désactive l'interface wlan0.
- **iwconfig wlan0 mode Ad-Hoc** : configure l'interface wlan0 en mode ad hoc.
- **iwconfig wlan0 essid 'adhoc'** : attribut un nom au réseau ad hoc.
- **ip link set wlan0 up** : active l'interface wlan0.
- **iwconfig wlan0 channel 2** : Définit le canal/fréquence à utiliser.
- **route -n** : affiche la table de routage IP avec les adresses numériques des hôtes voisins.
- **ping 192.168.0.2** : ping l'adresse IP 192.168.0.2 de l'hôte voisin pour s'assurer que la communication en mode ad hoc fonctionne.

Une fois que ce script est exécuté, le réseau ad hoc sera créé et la communication entre les deux ordinateurs est possible.

### 5.2.1.2 Installation d'OLSRD

Le protocole de routage ad hoc le plus utilisé dans le monde de l'open source est OLSRD de *olsr.org*. OLSRD-0.6.6.2 est la version installée dans ce testbed [120]. Cette version peut être téléchargée et installée soit via l'interface graphique Linux, soit via le terminal (voir Annexe A.2). Nous avons choisi cette implémentation car son code a été éprouvé, il est commenté et l'équipe avait déjà travaillé dessus.

La configuration d'exécution du démon OLSRD est décrite dans le fichier *olsrd.conf* qui se trouve par défaut dans le répertoire */etc*. Ce fichier de configuration contient des commentaires, des options simples et des blocs d'options.

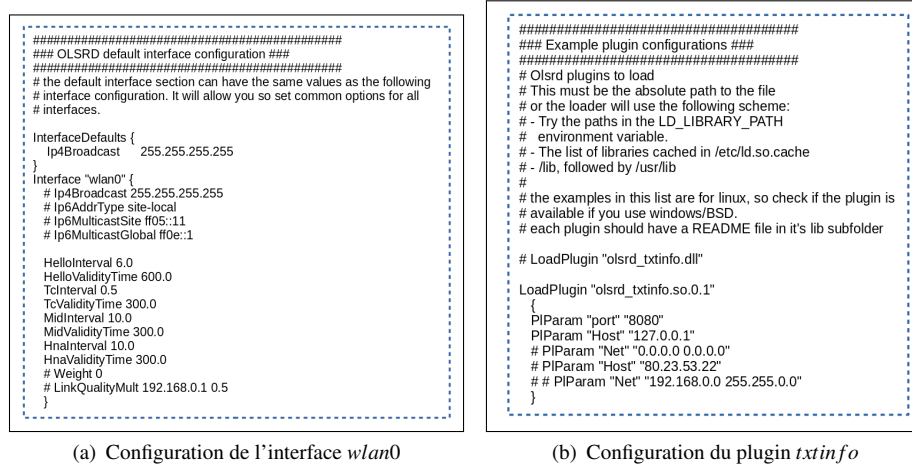


FIGURE 5.2 – Configuration d'OLSRD à partir du fichier *olsrd.conf*

Avant de démarrer OLSRD, il est nécessaire de s'assurer qu'il existe une interface active, par exemple *wlan0*. Ensuite, cette interface doit être configurée dans le fichier *olsrd.conf*, en spécifiant différents paramètres tels que : l'intervalle des messages HELLO, leurs temps de validité, l'intervalle des messages TC, etc. La configuration de *wlan0* permet donc de définir des options communes à toutes les interfaces. La figure 5.2 illustre un exemple de configuration simple d'OLSRD à partir du fichier *olsrd.conf*. La configuration de l'interface *wlan0* est décrite dans la figure 5.2 (a). Tandis que, la configuration du plugin *txtinfo* est illustrée dans la figure 5.2 (b). Ce plugin est très pratique, il permet d'extraire des informations d'OLSRD et il offre une vue de ces informations sur le net. Le plugin *txtinfo* va écouter le port 8080 et accepter les connexions depuis local host. Pour observer les différentes informations, il suffit :

- soit de se connecter sur l'URL : "http://127.0.0.1:8080/all",
- soit de taper sur un terminal la commande suivante : "echo "/all" | nc localhost 8080".

### 5.2.1.3 Implémentation d'ETX\_ANT dans OLSRD

La mise en oeuvre de ETX\_ANT est basée sur le protocole OLSRD. Après la réception de messages HELLO, OLSRD met à jour sa table de détection de lien. Cette fonction est appelée par le parser des messages HELLO, pour s'assurer que l'entrée en question (noeud voisin) soit créée ou mise à jour. À ce niveau, à chaque mise à jour ou création d'une entrée dans la détection de lien, nous mettons à jour la puissance du signal reçu pour chaque entrée. À cette fin, nous avons rajouté une structure à celle décrivant un lien ("struct link\_entry" dans "link\_set.h") et une fonction à celle responsable de la mise à jour des liens ("olsr\_update\_signal\_hello()" dans "link\_set.c"). Nous recevons la puissance du signal par la commande Linux "iw" qui permet d'afficher et de manipuler les appareils sans fil et leur configuration. Ensuite, lors du calcul de la métrique ETX pour un lien donné, il suffit de contrôler la puissance du signal reçu. Nous avons donc modifié les deux fichiers "link\_set.c" et "lq\_plugin\_default\_ff.c" en particulier les fonctions "update\_link\_entry()" et "default\_lq\_ff\_timer()". Nous avons également rajouté deux fonctions "signalPrediction()" et "default\_lq\_etx\_ant\_get\_fer()" qui prédise la puissance du signal et effectue la correspondance entre cette puissance et le FER. Les fichiers que nous avons modifié sont disponibles à [123]. Si la puissance du signal reçu est au-dessus du seuil limite noté  $TH_Q$  nous utilisons la métrique classique ETX. Dans le cas contraire, la puissance du signal reçu est prédite par l'algorithme de régression linéaire. Puis, nous faisons la correspondance entre la puissance du signal reçu estimée et la future valeur du FER. Dans le paragraphe suivant nous détaillons la méthode appliquée pour le mapping entre FER et la puissance du signal reçu.

#### 5.2.1.3.1 Mapping FER < -- > Signal

Pour le calcul du mapping entre FER et Signal, nous avons procédé comme suit :

- Nous avons configuré deux Pcs en mode ad hoc et qui acceptent les pings en broadcast. Pour ce faire, il suffit de taper la commande suivante en tant que root sur les 2 Pcs :

---

```
$echo 0 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
```

---

Le premier PC est mobile (Client avec une adresse IP 192.168.0.1) et le deuxième reste fixe (Serveur avec une adresse IP 192.168.0.2 et adresse MAC 7C:E9:D3:66:9E:B7).

Sur le PC mobile, nous lançons deux consoles :

La première console permet de surveiller la puissance du signal (instantanée et moyenne) sur l'interface wlan0 en spécifiant l'adresse MAC du serveur avec la commande suivante :

---

```
$watch -d 'iw wlan0 station get 7C:E9:D3:66:9E:B7|grep "signal"'
```

---



La deuxième console permet de lancer un ping en broadcast avec la commande :

---

```
$ping -c 100 -b 192.168.0.255
```

---

Le ping nous permet d'avoir une idée sur le taux de perte de la liaison sans fil. Il existe aussi un autre outil appelé iperf (voir [A.3](#)) pour mesurer le taux de perte entre client/serveur.

- Ensuite, nous déplaçons le PC mobile jusqu'à ce que nous commençons à voir des pertes (voir la figure [5.3](#)). À ce moment-là, nous collectons la correspondance entre la puissance du signal et le taux de perte équivalent. Nous avons effectué les mesures au sein de notre laboratoire (environnement indoor). Pour chaque puissance du signal, nous répétons cette méthode plusieurs fois pour avoir une valeur plus précise du taux de perte. Nous effectuons une correspondance entre la valeur de puissance du signal et la moyenne des tests des taux de perte mesurés. Nous continuons à éloigner le PC mobile du PC fixe jusqu'à ce que le taux de perte soit proche de 100%.
- Après nous répétons les mêmes tests mais en unicast, en tapant sur la deuxième console la commande suivante :

---

```
$ping -c 100 192.168.0.2
```

---

Nous collectons aussi les valeurs de la puissance du signal et le taux de perte correspondant.

- A la fin, nous avons un tableau récapitulatif avec les puissances du signal reçu et les taux de perte en unicast et broadcast.

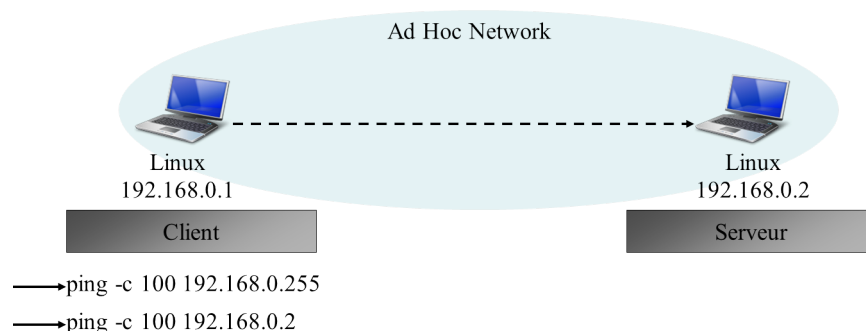


FIGURE 5.3 – Ping en broadcast et en unicast pour le Mapping FER < – > Signal

Les deux tableaux [5.2](#) et [5.3](#) reflètent respectivement un exemple de mapping entre FER et Signal en broadcast et unicast. Les mesures ont été effectués au sein de notre laboratoire, là où nous allons effectuer les tests finaux du testbed.

Nous pouvons déduire les valeurs de perte pour chaque puissance, en utilisant la correspondance FER et signal en broadcast et unicast mesurés et affichés dans les deux tableaux [5.2](#) et [5.3](#).

Puissance du signal	-- >	Packet loss
Entre -70 et -75 dBm	-- >	packet loss = 6%
Entre -76 et -80dBm	-- >	packet loss= 11%
Entre -81 et -85dBm	-- >	packet loss= 35%
à -87 dBm	-- >	packet loss = 58%
à -88 dBm	-- >	packet loss= 81%
entre -89dBm et Plus	-- >	packet loss = 100%

TABLE 5.2 – Correspondance FER &amp; Signal en broadcast

Puissance du signal	-- >	Packet loss
Entre -70 et -75 dBm	-- >	packet loss = 1%
Entre -76 et -80dBm	-- >	packet loss= 2%
Entre -81 et -85dBm	-- >	packet loss= 4%
à -87 dBm	-- >	packet loss = 38%
à -88 dBm	-- >	packet loss= 42%
à -89 dBm	-- >	packet loss= 70%
< -89dBm	-- >	packet loss = 100%

TABLE 5.3 – Correspondance FER &amp; Signal en unicast

Puissance du signal	-- >	Packet loss
Entre -70 et -75 dBm	-- >	P = 5.52%
Entre -76 et -80dBm	-- >	P = 9.1%
Entre -81 et -85dBm	-- >	P = 32%
à -87 dBm	-- >	P = 46%
à -88 dBm	-- >	P = 75%
à -89 dBm	-- >	P = 94%
< -89dBm	-- >	P = 100%

TABLE 5.4 – Correspondance FER &amp; Signal

Pour ce faire, nous notons le taux de perte mesuré en unicast par *proba\_perte\_unicast\_mesure*, le taux de perte mesuré en broadcast par *proba\_perte\_broadcast\_mesure*. Pour déterminer la perte en unicast noté  $p_{unicast}$ , nous avons utilisé l'équation suivante :

$$p_{unicast} = 1 - \sqrt{(1 - proba\_perte\_unicast\_mesure)} \quad (.5.2.1)$$

D'où les valeurs de perte pour chaque puissance notée  $P$ , sont déduites par :

$$p = 1 - \frac{(1 - proba\_perte\_broadcast\_mesure)}{(1 - p_{unicast})} \quad (.5.2.2)$$

Après le calcul de ces équations, nous obtenons une correspondance des valeurs de perte pour chaque puissance du signal dans le tableau 5.4. Ce mapping est utilisé dans le code d'OLSRD, pour correspondre la bonne valeur du FER pour une puissance du signal mesurée.



Parameters	Numerical Values
Wireless technology	IEEE 802.11 bgn
Number of nodes	5
Routing Protocol	OLSRD 0.6.6.2
Frequency	2.412 GHz
Number of packet for low walking	100 packets
Number of packet for fast walking	60 packets
Packets sent by	192.168.0.1
Packets received by	192.168.0.2
Network protocol analyzer	Ping
ANTICIPATION_TIME (OLSRD)	5 sec
TH_Q (OLSRD)	-64 dBm
HELLO_INTERVAL (OLSRD)	0.25, 0.5, 1, 2 sec
TC_INTERVAL (OLSRD)	2, 5 sec

TABLE 5.5 – Experimental Parameters

le noeud fixe  $N2$ . Au départ,  $N2$  est à portée radio de  $N1$  et  $N3$ . Et  $N4$  est à portée radio de  $N3$  et  $N5$ . Ensuite nous déplaçons le noeud mobile  $N1$  tout au long du couloir pour passer d'une salle à une autre jusqu'à ce qu'il atteigne la fin de la chaîne le noeud  $N5$ . Pour faire bouger  $N1$ , nous utilisons deux rythmes de marche : lente et rapide. La marche lente peut refléter le déplacement des utilisateurs avec leur unité portable. Par contre, la marche rapide peut refléter un utilisateur sur un vélo. Nous n'avons pas pu utiliser un autre moyen de mobilité vue le manque de moyens pour ce testbed.

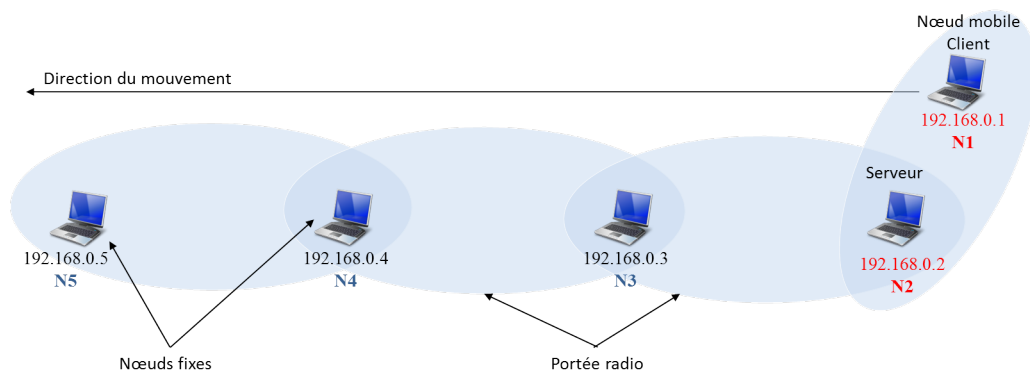


FIGURE 5.5 – Voisinage des noeuds du testbed

### 5.3 Résultats et discussion

Nous comparons les performances des trois métriques Hop Count, ETX et ETX\_ANT implémentées dans le protocole OLSRD. Nous étudions le comportement du protocole OLSRD en

Métrique HELLO_INTERVAL	TC_INTERVAL = 2 sec			TC_INTERVAL = 5 sec		
	Hop Count	ETX	ETX_ANT	Hop Count	ETX	ETX_ANT
0.25	31%	22%	8%	48%	24%	9%
0.5	33%	23%	9%	50%	26%	10%
1	35%	26%	11%	53%	29%	13%
2	49%	29%	13%	57%	34%	15%

TABLE 5.6 – Packet loss des différentes métriques (marche lente)

Métrique HELLO_INTERVAL	TC_INTERVAL = 2 sec			TC_INTERVAL = 5 sec		
	Hop Count	ETX	ETX_ANT	Hop Count	ETX	ETX_ANT
0.25	40%	25%	10%	51%	31%	12%
0.5	44%	27%	12%	54%	32%	15%
1	47%	30%	14%	56%	35%	17%
2	53%	33%	17%	60%	37%	19%

TABLE 5.7 – Packet loss des différentes métriques (marche rapide)

terme du taux de perte et de la sélection de route pour chaque métrique. Les paramètres de l'expérimentation sont présentés dans le tableau 5.5. Tous les réglages des paramètres ont été effectuées en utilisant des lignes de commande Shell (sur un terminal). Avant de démarrer OLSRD, nous avons fixé HELLO\_INTERVAL, TC\_INTERVAL et la métrique à utiliser, dans le fichier *olsrd.conf*. Nous avons utilisé deux valeurs pour TC\_INTERVAL (2s et 5 sec) et nous avons varié HELLO\_INTERVAL (de 0.25, à 2 sec). Pour utiliser la métrique Hop Count, il suffit de l'activer avec LinkQualityLevel 0. La métrique ETX est sélectionnée par activation de l'algorithme de qualité de lien à "etx\_ff" avec LinkQualityAlgorithm "etx\_ff", si nous désirons utiliser la métrique "etx\_ff". Pour chaque configuration, nous avons effectué 10 tests, ensuite, nous faisons la moyenne pour avoir la mesure correspondante.

Les résultats du testbed sont présenté dans les deux tableaux 5.6 et 5.7. Le tableau 5.6 montre les taux perte des différentes métriques avec une faible mobilité (marche lente). Le tableau 5.7 montre aussi les taux de perte des différentes métriques mais avec une mobilité plus importante (marche rapide).

La figure 5.6 illustre les taux de perte en fonction du HELLO\_INTERVAL pour chaque métrique avec TC\_INTERVAL = 2 sec. Nous observons que les taux de perte des trois métriques augmentent avec la mobilité. Mais, le taux de perte d'ETX\_ANT reste inférieur à celui d'Hop Count et ETX quel que soit le rythme de marche utilisée. Avec une marche lente, il ne dépasse pas 13%, et il atteint 17% avec une marche rapide. Hop Count est la pire métrique, son taux de perte dépasse celui d'ETX, et il atteint 53%. Nous retrouvons donc la hiérarchie obtenue lors des simulations.

Nous remarquons aussi dans la figure 5.6, que la fréquence des messages HELLO a une influence sur le taux de perte des métriques. En effet, packet loss des trois métriques augmente lorsque nous diminuons la fréquence des HELLO. Par exemple pour ETX\_ANT (marche lente), avec HELLO\_INTERVAL= 0.25 sec, packet loss ne dépasse pas 8%, mais il atteint 13% avec HELLO\_INTERVAL= 2 sec. Cela s'explique par le fait qu'une augmentation dans la fréquence des HELLO permet une mise à jour plus fréquente des tables de routage, ce qui en résulte moins de perte.

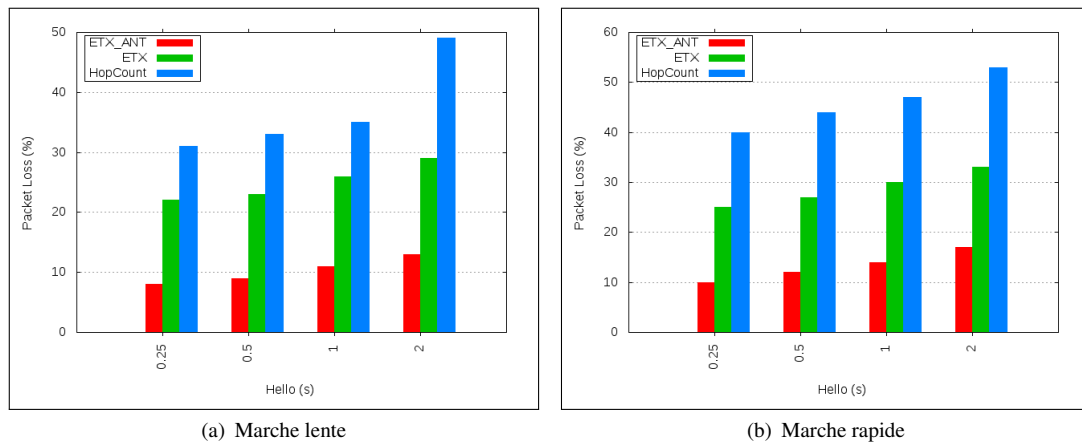


FIGURE 5.6 – Packet loss en fonction des Hellos pour Hop Count, ETX et ETX\_ANT (TC = 2s)

La figure 5.7 illustre les taux de perte en fonction du HELLO\_INTERVAL pour chaque métrique avec TC\_INTERVAL = 5 sec. Les résultats quantitatifs sont similaires à la configuration précédente (TC\_INTERVAL = 2 sec). Packet loss avec ETX\_ANT est inférieur à celui d'ETX et Hop Count quel que soit le rythme de marche. Il atteint 15% pour une marche lente (voir figure 5.7 (a)) et 19% pour une marche rapide (voir figure 5.7 (b)). Hop count présente un taux de perte très élevé par rapport à ETX, surtout avec une marche rapide, il dépasse 59%.

Nous remarquons aussi dans la figure 5.7 que les taux de pertes des différentes métriques avec TC\_INTERVAL = 5 sec est plus grand qu'avec TC\_INTERVAL = 2 sec. Cela se traduit par l'influence de la fréquence des messages TC dans le routage. Lorsque les messages TC sont plus fréquents dans le réseau, les tables de routage sont mises à jour plus fréquemment, et de cette manière, OLSRD peut maintenir des routes plus stables.

Après avoir étudié le comportement d'OLSRD en terme du taux de perte, nous passons à analyser son comportement dans la sélection de route pour chaque métrique. Pour ce faire, lors de l'exécution d'OLSRD, nous avons lancé sur un terminal la commande :

---

```
$tracert 192.168.0.2
```

---

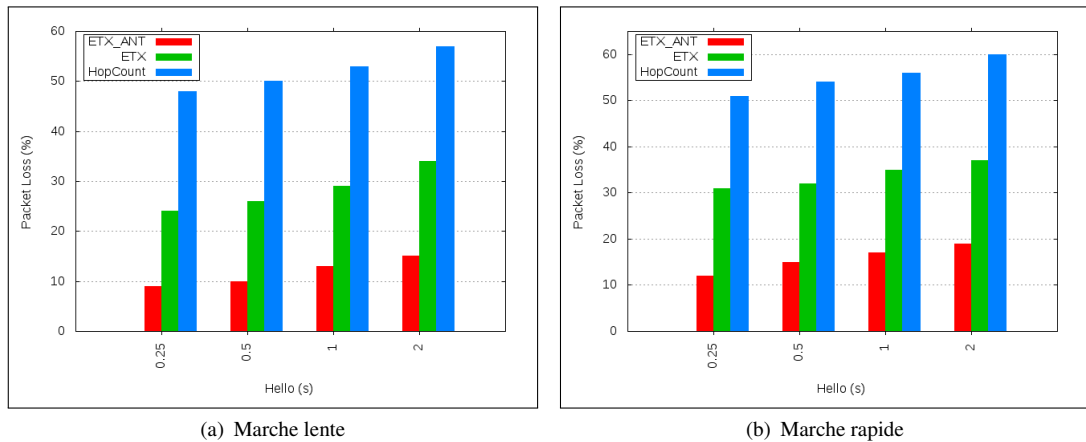


FIGURE 5.7 – Packet loss en fonction des Hellos pour Hop Count, ETX et ETX\_ANT (TC = 5s)

Cette commande permet de tracer la route vers une destination. Nous avons utilisé cette commande plusieurs fois, dans chaque expérimentation et pour chaque configuration. La figure 5.8 montre un exemple d'exécution de cette commande.

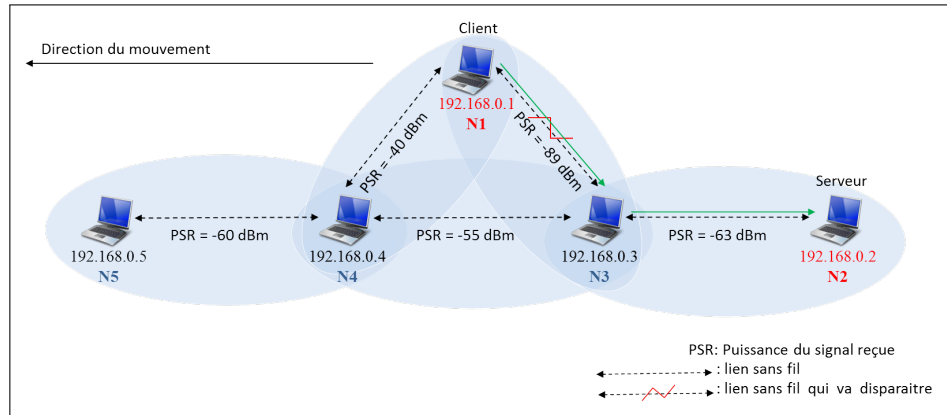
```

root@sabrine-VPCEA36FF:/usr/local/src/olsrd-0.6.6.2$ traceroute 192.168.0.2
traceroute to 192.168.0.2 (192.168.0.2), 64 hops max
 1 192.168.0.3 2.027ms 1.828ms 1.942ms
 2 192.168.0.2 1.941ms 1.848ms 1.888ms
root@sabrine-VPCEA36FF:/usr/local/src/olsrd-0.6.6.2$ traceroute 192.168.0.2
traceroute to 192.168.0.2 (192.168.0.2), 64 hops max
 1 192.168.0.2 35.697ms 7.045ms 7.261ms
root@sabrine-VPCEA36FF:/usr/local/src/olsrd-0.6.6.2$ traceroute 192.168.0.2
traceroute to 192.168.0.2 (192.168.0.2), 64 hops max
 1 192.168.0.2 2.113ms 2.859ms 2.154ms
root@sabrine-VPCEA36FF:/usr/local/src/olsrd-0.6.6.2$ traceroute 192.168.0.2
traceroute to 192.168.0.2 (192.168.0.2), 64 hops max
 1 192.168.0.2 47.614ms 63.400ms 9.871ms
root@sabrine-VPCEA36FF:/usr/local/src/olsrd-0.6.6.2$ traceroute 192.168.0.2
traceroute to 192.168.0.2 (192.168.0.2), 64 hops max
 1 192.168.0.3 2.537ms 12.586ms 4.577ms
 2 192.168.0.2 8.279ms 3.654ms 3.479ms
root@sabrine-VPCEA36FF:/usr/local/src/olsrd-0.6.6.2$ traceroute 192.168.0.2
traceroute to 192.168.0.2 (192.168.0.2), 64 hops max
 1 192.168.0.2 2.285ms 2.303ms 1.675ms
 2 192.168.0.2 5.402ms 3.115ms 1.691ms
root@sabrine-VPCEA36FF:/usr/local/src/olsrd-0.6.6.2$ traceroute 192.168.0.2
traceroute to 192.168.0.2 (192.168.0.2), 64 hops max
 1 192.168.0.2 6.291ms 13.420ms 13.293ms
root@sabrine-VPCEA36FF:/usr/local/src/olsrd-0.6.6.2$ traceroute 192.168.0.2
traceroute to 192.168.0.2 (192.168.0.2), 64 hops max
 1 192.168.0.4 10.596ms 4.040ms 100.631ms
 2 192.168.0.3 4.376ms 3.160ms 2.462ms
 3 192.168.0.2 4.309ms 6.168ms 83.413ms
root@sabrine-VPCEA36FF:/usr/local/src/olsrd-0.6.6.2$ traceroute 192.168.0.2
traceroute to 192.168.0.2 (192.168.0.2), 64 hops max
 1 192.168.0.2 664.256ms 247.631ms 2.656ms
root@sabrine-VPCEA36FF:/usr/local/src/olsrd-0.6.6.2$ traceroute 192.168.0.2
traceroute to 192.168.0.2 (192.168.0.2), 64 hops max
 1 192.168.0.2 4.908ms 3.528ms 2.573ms
root@sabrine-VPCEA36FF:/usr/local/src/olsrd-0.6.6.2$ traceroute 192.168.0.2

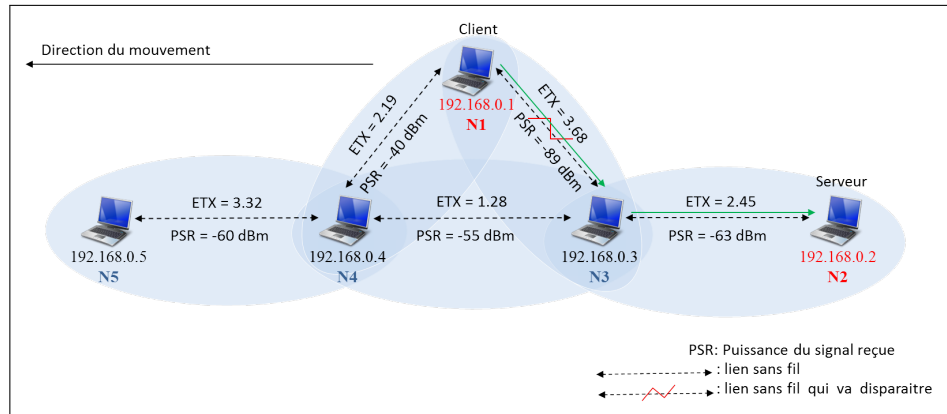
```

FIGURE 5.8 – Affichage de route par la commande “traceroute”

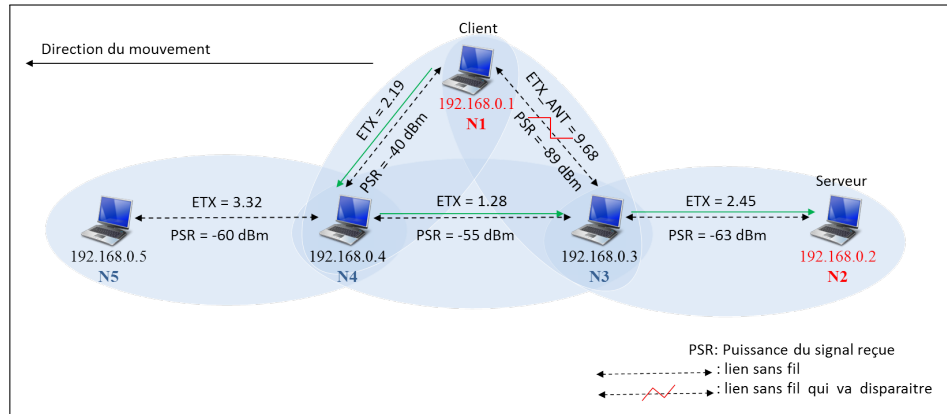
La figure 5.9 affiche les routes sélectionnées pour chaque métrique, lorsque N1 est à portée radio de N3 et N4. N1 bouge en direction de N5, c’est-à-dire la liaison entre N1 et N3 à tendance à disparaître. Ce lien est un lien défectueux avec une grande probabilité de perte. Basé sur la métrique Hop Count, OLSRD sélectionne simplement la première route avec un nombre de saut minimale de N1 vers N2. Dans ce cas, la route sélectionnée est  $N1 \rightarrow N3 \rightarrow N2$ , comme l’illustre la figure 5.9 (a). Basé sur ETX, le choix de la route prend en compte la qualité de la liaison de tous les liens de la route. Le chemin  $N1 \rightarrow N3 \rightarrow N2$  offre un coût plus petit que le chemin  $N1 \rightarrow N4 \rightarrow N3 \rightarrow N2$ . Pour cette raison,  $N1 \rightarrow N3 \rightarrow N2$  est la route sélectionnée par ETX (voir figure 5.9 (b)).



(a) Hop Count



(b) ETX



(c) ETX\_ANT

FIGURE 5.9 – Sélection de route pour chaque métrique

Sur la base d'ETX\_ANT, OLSRD considère, en plus de la qualité de la liaison, la puissance du signal reçu pour chaque liaison sans fil qui diffère d'un lien vers un autre dans nos expériences. Comme le montre la figure 5.9 (c), la route  $N1 \rightarrow N4 \rightarrow N3 \rightarrow N2$  inclut des liens plus stables et plus pertinents. Nous observons dans la figure 5.9 (c), que la puissance de signal entre N1 et N3 atteint -89 dBm, dans ce cas, la métrique ETX\_ANT gonfle la métrique ETX de telle sorte que ce lien ne sera plus considéré dans la sélection de route. De cette manière, le coût du chemin  $N1 \rightarrow N3 \rightarrow N2$  devient plus grand que le coût du chemin  $N1 \rightarrow N4 \rightarrow N3 \rightarrow N2$ . Donc, la



route sélectionnée par ETX\_ANT de N1 vers N2 est  $N1 \rightarrow N4 \rightarrow N3 \rightarrow N2$ .

## 5.4 Conclusion

Au cours de ce chapitre, nous avons mis en oeuvre la métrique ETX\_ANT dans notre testbed basé sur le protocole de routage OLSRD. Nous avons mené des expériences pour comparer les performances de Hop Count, ETX et ETX\_ANT. Nous avons changé plusieurs paramètres tels que l'intervalle des messages HELLO, l'intervalle des messages TC et la mobilité pour comparer l'effet de variation de ces paramètres dans les performances des métriques. Nous avons évalué les performances du testbed en terme du taux de perte dans nos expériences, et nous avons trouvé les résultats suivants :

- Lorsque l'intervalle des messages HELLO augmente, le taux de perte des trois métriques augmente, car la mise à jour des tables de routage diminue quand la surcharge des messages HELLO diminue.
- Lorsque l'intervalle des messages TC augmente, le taux de perte des trois métriques augmente aussi, étant donné que l'augmentation de la charge des messages TC rend la mise à jour des routes plus fréquentes.
- Le taux de perte de Hop Count, ETX et ETX\_ANT augmente avec la mobilité (marche lente, marche rapide), mais les performances d'ETX\_ANT surpassent celles des autres métriques quelle que soit la configuration utilisée dans le testbed. Le taux de perte d'ETX\_ANT ne dépasse pas 19%. Par contre, le taux de perte avec ETX atteint 37% et avec Hop Count atteint 60%. Nous n'avons pas atteint 0% de perte comme dans les simulations, mais ceci est sans doute à plusieurs facteurs :
  1. l'environnement indoor est particulièrement défavorable à notre test car il y a une croissance très rapide des pertes,
  2. l'environnement radio est très instable géographiquement,
  3. et les cartes Wi-Fi étaient internes aux Pcs (pas d'antennes externes).

Un dimensionnement plus fin aurait pu être fait en environnement outdoor et la croissance du taux de perte aurait été plus stable et plus lente qu'en indoor.

# Conclusion & perspectives

Au cours de cette thèse, nous avons proposé en premier lieu deux nouveaux algorithmes de calcul des métriques de routage dans les MANET. L'approche consistant à anticiper leur valeur, et à notre connaissance originale dans le contexte des réseaux ad hoc. Ces algorithmes sont basés sur le calcul des métriques classiques ETX et ETT. Ensuite, nous avons validé les performances de notre approche par rapport aux métriques classiques dans un testbed.

Dans un premier temps, nous nous sommes intéressés au problème de mobilité avec les métriques classiques Hop count, ETX, ETT et LD. Dans presque tous les travaux que nous avons étudié, Hop Count est la métrique la plus utilisée par les protocoles de routage dans les MANET. Par contre, ETX et ETT sont considérées comme les métriques de base qui sont capables de refléter la qualité des liens. LD est estimée comme une métrique sensible à la mobilité. Cependant, les simulations effectuées pour différentes topologies ont montré que les performances des métriques classiques s'écroulent avec la mobilité. Cette dégradation de performances en présence de la mobilité est due à un décalage entre les valeurs réelles de ces métriques et celles prises en compte dans le processus de calcul de route.

Dans la suite, nous avons pu améliorer les performances de ces métriques de qualité de lien ETX et ETT, en présence de la mobilité, en proposant deux algorithmes d'anticipation de ces métriques. L'idée de base est de prédire la valeur de la métrique un certain temps à l'avance en utilisant des mesures de qualité des couches physiques et MAC. Un mapping reliant la qualité du lien mesurée à la couche physique et le FER est utilisé dans les deux algorithmes. Ces deux derniers diffèrent dans le système de prédiction. Le premier algorithme est basé sur un développement de Taylor. Le deuxième algorithme est basé sur une régression linéaire. La validation de ces nouvelles métriques a été effectuée par un grand nombre de simulations qui ont été évaluées au travers de trois métriques de performance : le PDR, la capacité et le délai de bout en bout. Trois types de topologie ont été simulées, avec et sans fading. Les résultats de simulation ont montré que le routage basé sur ces nouvelles métriques reste stable lorsque les paramètres de ces algorithmes sont correctement dimensionnés. L'algorithme basé sur la régression linéaire est systématiquement meilleur que l'algorithme basé sur le développement de Taylor. L'approche considérée est donc efficace et permet de bien gérer la mobilité. Ses performances sont donc très logiquement liées à l'efficacité de la méthode de prédiction.

La dernière partie de cette thèse consistait à développer notre approche dans un testbed. Cette technique nous a permis de tester les performances des métriques classiques et des métriques anticipées. Les résultats des tests du testbed confirment l'efficacité de notre approche. Les performances de la métrique anticipée sont meilleures que celles des métriques classiques. C'est vrai que dans ce testbed, nous avons adopté uniquement la topologie chaîne avec différent rythme de marche : lente et rapide. Mais, dans la continuité des travaux de cette thèse, nous pourrions étudier l'impact de la mobilité avec d'autres topologies possibles et d'autres types de mobilité. Nous pensons aussi à étudier le mapping du débit pour la métrique ETT\_ANT, et tester ces performances dans le testbed. Une extension possible de ce travail, actuellement concentrée sur les MANETs, serait de l'utiliser dans d'autres contextes. En particulier, il serait intéressant de l'adapter pour les équipements personnels type smartphone, où ces métriques appliquées aux différentes interfaces radio permettraient de basculer de manière transparente d'une technique à l'autre.

## Annexe A

# ANNEXE A

### A.1 Configuration en mode ad hoc en utilisant Network Manager

Il existe deux étapes dans cette méthode de mise en place d'un réseau en mode ad hoc. La première étape consiste à spécifier les paramètres qui vont permettre la création du réseau. Une fois ces paramètres sont enregistrés, l'accès au réseau est faisable à n'importe quel moment. La deuxième étape est la connexion au réseau à chaque fois que la machine désire être membre du réseau ad hoc.

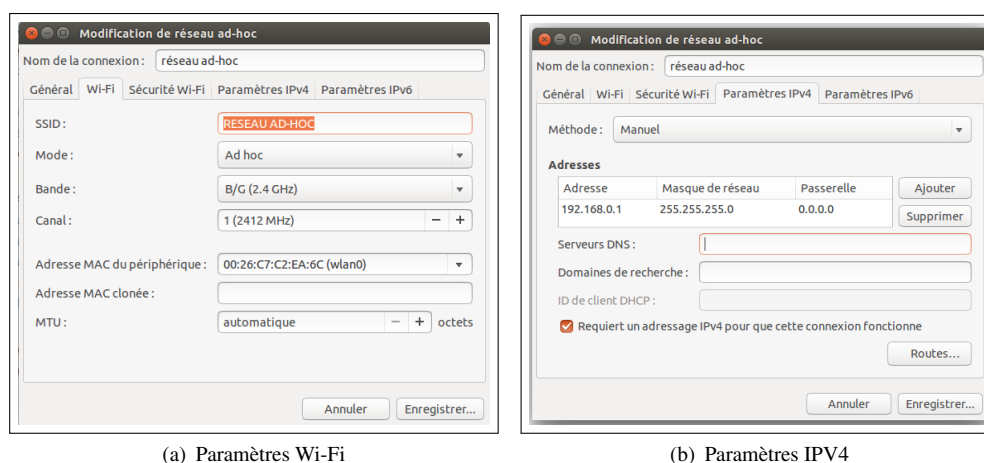


FIGURE A.1 – Configuration en mode ad hoc en utilisant Network Manager

Le Network Manager permet de créer une nouvelle connexion Wi-Fi en mode ad hoc à partir de l'onglet "Modification des connexions". La figure A.1(a) montre les différents champs à remplir pour configurer cette connexion. La figure A.1(b) présente les paramètres "Wi-Fi" tel que :

- SSID (Service Set Identifier) : présente le nom du réseau,

- Mode : spécifie le mode de connexion soit en mode “infrastructure” ou en mode “ad hoc”. Dans notre cas, nous sélectionnons le mode “ad hoc”.
- Bande : définit la bande passante sur laquelle doit travailler le réseau ad hoc,
- Canal : présente le canal de connexion,
- Adresse MAC : présente le code d’identification d’un périphérique. Chaque périphérique connecté à un réseau possède une adresse MAC unique (attribuée en usine).

Après avoir configuré les paramètres “Wi-Fi”, nous passons aux “Paramètres IPv4 ”(voir figure A.1.b) tel que :

- Méthode : il existe différentes possibilités de définir cette information. Elle peut être spécifiée à : Automatique (DHCP) ou Adresses automatiques uniquement (DHCP) ou Manuel ou Lien-local uniquement ou Partagé avec d’autres ordinateurs ou Désactivé. Dans notre cas, la méthode est définie comme manuelle.
- Adresse : 192.168.0.1 présente l’adresse IP de l’ordinateur actuel, il faut attribuer une différente par ordinateur.
- Masque de réseau : 255.255.255.0 est le masque de réseau à utiliser pour tous les ordinateurs du réseau.
- Passerelle : mis à 0.0.0.0 signifie aucune mais dans le cas où un ordinateur sera une passerelle vers un autre réseau il suffit de mettre son adresse IP qui fera office de passerelle.

Une fois que ces paramètres sont enregistrés, un nouveau réseau ad hoc est créé. Il suffit alors de choisir le nom du réseau ad hoc avec le Network Manager, en sélectionnant “Se connecter à un réseau sans fil invisible ”. Un ping peut être utilisé pour tester la connexion entre les deux ordinateurs.

## A.2 Installation d’OLSRD

Une fois OLSRD est téléchargé, les commandes suivantes [124] permettent l’installation et la compilation de ce protocole sur un terminal :

---

```
$cp olsrd-0.6.6.2.tar.bz2 /usr/local/src/  
$cd /usr/local/src  
$tar jxvf olsrd-0.6.6.2.tar.bz2  
$cd /usr/local/src/olsrd-0.6.7.2  
$make all  
$make install  
$make libs  
$make install\_libs  
$mkdir -p /etc  
$cp -i files/olsrd.conf.default.lq /etc/olsrd.conf
```

---

### A.3 Iperf

Iperf [125] est un outil de test de performance de réseau fiable. Cet outil est open source qui permet de fournir des statistiques sur la bande passante et la qualité d'un lien réseau à l'aide de différentes propriétés des protocoles : **T**ransmission **C**ontrol **P**rotocol (**TCP**) et **U**ser **D**atagram **P**rotocol (**UDP**).

La bande passante est mesurée par des tests iperf TCP. Quant à la qualité d'un lien, elle est déterminée principalement par les facteurs suivants :

- Latence (temps de réponse ou RTT) : mesurée à l'aide d'un ping.
- Gigue (variation de la latence) ou Jitter : mesurée par un test iperf UDP.
- Perte de paquet : mesurée avec un test Iperf UDP.

Iperf peut être installé très facilement sur n'importe quel système UNIX/Linux ou Microsoft Windows. Un hôte doit être configuré en tant que client et l'autre en tant que serveur. Ce dernier est lancé en premier. La figure A.2 montre un exemple où iperf est installé sur deux machines Linux. L'une est utilisée comme client iperf et l'autre comme serveur Iperf.

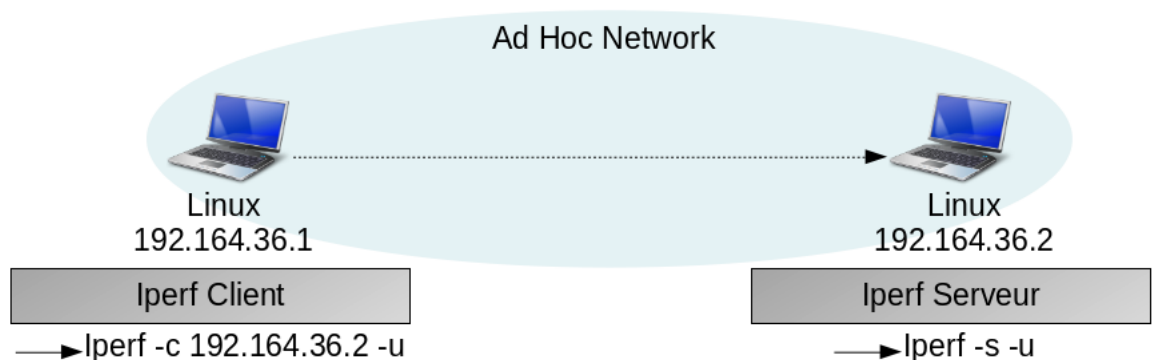


FIGURE A.2 – Configuration iperf entre client/serveur

**Côté serveur :** *iperf -s -u*

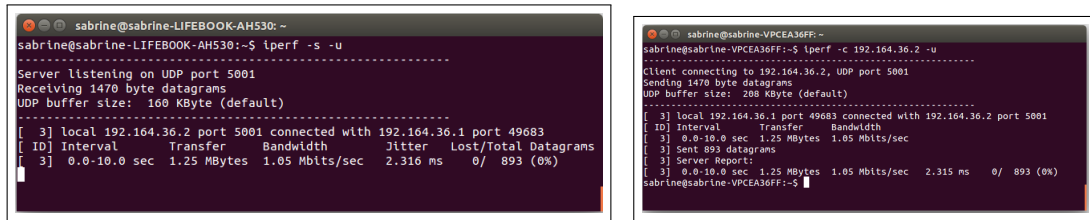
L'option *-s* indique l'hôte serveur, et l'option *-u* désigne le flux UDP (voir figure A.3(a)).

**Côté client :** *iperf -c 192.164.36.2 -u*

L'option *-c* indique l'hôte client. 192.164.36.2 désigne l'adresse IP du serveur (voir figure A.3(b)).

Après avoir exécuté le test du client au serveur, la commande iperf sur les deux htes (voir A.3) retourne diverses informations :

- **Interval** : spécifie la durée du test en seconde, par défaut 10 secondes.
- **Transfer** : indique le taux de transfert exprimé en Mo.



```

sabrine@sabrine-LIFEB00K-AH530: ~
sabrine@sabrine-LIFEB00K-AH530:~$ iperf -s -u
.....
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 160 KByte (default)
.....
[ 3] local 192.164.36.2 port 5001 connected with 192.164.36.1 port 49683
[ ID] Interval      Transfer    Bandwidth    Jitter    Lost/Total Datagrams
[ 3] 0.0-10.0 sec  1.25 MBytes  1.05 Mbits/sec  2.316 ms    0/ 893 (0%)

```

```

sabrine@sabrine-VPCEA36FF: ~
sabrine@sabrine-VPCEA36FF:~$ iperf -c 192.164.36.2 -u
.....
Client connecting to 192.164.36.2, UDP port 5001
Sending 1470 byte datagrams
UDP buffer size: 208 KByte (default)
.....
[ 3] local 192.164.36.1 port 49683 connected with 192.164.36.2 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 3] 0.0-10.0 sec  1.25 MBytes  1.05 Mbits/sec
[ 3] Sent 893 datagrams
[ 3] Server Report:
[ 3] 0.0-10.0 sec  1.25 MBytes  1.05 Mbits/sec  2.315 ms    0/ 893 (0%)
sabrine@sabrine-VPCEA36FF:~$

```

(a) Iperf côté serveur

(b) Iperf côté client

FIGURE A.3 – Commandes iperf pour la mesure de performance entre client/serveur

- **Bandwidth** : mesure la bande passante exprimée en Mb.
- **Jitter** : spécifie le délai Jitter.
- **Lost/Total Datagram** : mesure le taux de perte des paquets exprimé en %.

Iperf est beaucoup plus fiable dans ses résultats de test par rapport à de nombreux autres outils de tests de réseau en ligne. En plus, cet outil présente un avantage supplémentaire dans son utilisation du fait qu'il peut mesurer les performances de deux serveurs à la fois même s'ils sont situés dans des régions géographiques différentes.

# Bibliographie

- [1] C. Perkins, D. Johnson, and J. Arkko. Mobility support in IPv6, RFC6275, jul 2011. URL <http://www.ietf.org/rfc/rfc6275.txt>.
- [2] H.M. Ali, A.M. Naimi, A. Busson, and V. Vèque. An efficient link management algorithm for high mobility mesh networks. In *Proceedings of the 5th ACM International Workshop on Mobility Management and Wireless Access*, MobiWac '07, pages 42–49, 2007.
- [3] D.S.J. De Couto, D. Aguayo, B.A. Chambers, and R. Morris. Performance of multihop wireless networks : Shortest path is not enough. *SIGCOMM Comput. Commun. Rev.*, 33 (1) :83–88, jan 2003. ISSN 0146-4833.
- [4] D.S.J. De Couto, D. Aguayo, J. Bicket, and R. Morris. A high-throughput path metric for multi-hop wireless routing. *Wirel. Netw.*, 11(4) :419–434, July 2005.
- [5] P.M. Esposito, Campista M.E.M., Moraes I.M., Costa L.H.M.K., C.M.B.D. Otto, and Rubinstein M.G. Implementing the expected transmission time metric for OLSR wireless mesh networks. *1st IFIP Wireless Days. WD '08.*, 24-27 Nov. 2008.
- [6] C. Yawut. *Adaptation à la mobilité dans les réseaux ad-hoc*. Thèse de doctorat, Institut National Polytechnique de Toulouse, laboratoire IRIT, Toulouse, France, octobre 2009.
- [7] M. BA. *Vers une structuration auto-stabilisante ad hoc : cas des réseaux de capteurs sans fil*. Thèse de doctorat, Université de Reims Champagne-Ardenne, Laboratoire CReSTIC, France, May 2014.
- [8] S. Basagni, M. Conti, and S. Giordano. *Mobile Ad Hoc Networking*. Wiley-IEEE Press, August 2004.
- [9] S. Pomportes. *Mobilité et gestion efficace des fréquences dans un réseau ad hoc à forte efficacité*. Thèse de doctorat, Supélec, Laboratoire des Signaux et Systèmes (L2S), Université Paris-Sud, décembre 2011.
- [10] T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. *Wireless Communications and Mobile Computing*, 2(5) :483–502, 2002.



- [11] Umang, B.V. R. Reddy, and M.N. Hoda. Study of mobility management schemes in mobile adhoc networks. *International Journal of Computer Applications*, 17(7) :42–47, March 2011.
- [12] D.B. Johnson and D.A. Maltz. Dynamic Source Routing in Ad Hoc Wireless Networks. *Mobile Computing*, pages 153–181, 1996.
- [13] S. Kurkowski, T. Camp, and M. Colagrosso. MANET simulation studies : The incredible. *ACM SIGMOBILE Mobile Computing and Communications Review*, 9 :50–61, 2005.
- [14] E.M. Royer, P.M. Melliar-Smith, and L.E. Moser. An analysis of the optimum node density for ad hoc mobile networks. In *Communications, 2001. ICC 2001. IEEE International Conference on*, volume 3, pages 857–861 vol.3, 2001.
- [15] B. Liang and Z.J. Haas. Predictive distance-based mobility management for multidimensional pcs networks. *IEEE/ACM Trans. Netw.*, 11(5) :718–732, October 2003.
- [16] R.R. Roy. *Handbook of Mobile Ad Hoc Networks for Mobility Models*. Springer-Verlag New York, 1st edition, 2010.
- [17] Beigh B.M. and Prof.M.A.Peer. Classification of current routing protocols for ad hoc networks - a review. *International Journal of Computer Applications*, 7(8) :26–32, October 2010.
- [18] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Dijkstra's algorithm*. MIT Press.
- [19] D. Bertsekas and R. Gallager. *Data Networks (2nd Ed.)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1992.
- [20] C.E. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. *SIGCOMM Comput. Commun. Rev.*, 24(4) :234–244, October 1994.
- [21] S. Murthy and J. J. Garcia-Luna-Aceves. An efficient routing protocol for wireless networks. *Mob. Netw. Appl.*, 1(2) :183–197, October 1996.
- [22] Tsu-Wei C. and M. Gerla. Global state routing : a new routing scheme for ad-hoc wireless networks. In *IEEE International Conference on Communications (ICC 98)*, volume 1, pages 171–175 vol.1, Jun 1998.
- [23] G. Pei, M. Gerla, and T.W. Chen. Fisheye state routing in mobile ad hoc networks. In *ICDCS Workshop on Wireless Networks and Mobile Computing*, pages 71–78, Jun 2000.

- [24] A. Iwata, C.C. Chiang, G. Pei, M. Gerla, and T.W. Chen. Scalable routing strategies for ad hoc wireless networks. volume 17, pages 1369–1379, Aug 1999.
- [25] M. Joa-Ng and I-Tai Lu. A peer-to-peer zone-based two-level link state routing for mobile ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 17(8) :1415–1425, Aug 1999.
- [26] C. Ching-Chuan and M. Gerla. Routing and multicast in multihop, mobile wireless networks. In *IEEE 6th International Conference on Universal Personal Communications Record*, volume 2, pages 546–551 vol.2, Oct 1997.
- [27] S. Basagni, I. Chlamtac, V.R. Syrotiuk, and B.A. Woodward. A distance routing effect algorithm for mobility (dream). In *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, MobiCom '98, pages 76–84, New York, NY, USA, 1998. ACM.
- [28] T. Clausen and P. Jacquet. Optimized link state routing protocol (OLSR), RFC3626, oct 2003. URL <http://www.ietf.org/rfc/rfc3626.txt>.
- [29] N. Badache, D. Djenourf, A. Derhab, and T. Lemlouma. Les protocoles de routage dans les réseaux mobiles ad hoc. *RIST : Revue d'Informatique Scientifique et Technique*, 12 (02) :77–112, 2002.
- [30] C. Perkins, E. Belding-Royer, and S. Das. Ad hoc on-demand distance vector (AODV) routing, RFC3561, 2003. URL <http://www.ietf.org/rfc/rfc3561.txt>.
- [31] S. Sharma and S.K. Jena. Cluster based multipath routing protocol for wireless sensor networks. *SIGCOMM Comput. Commun. Rev.*, 45(2) :14–20, April 2015.
- [32] M.S. Corson and A. Ephremides. A distributed routing algorithm for mobile wireless networks. *Wirel. Netw.*, 1(1) :61–81, February 1995.
- [33] V.D. Park and M.S. Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. In *Proceedings of the Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. INFOCOM '97.*, volume 3, pages 1405–1413, Washington, DC, USA, Apr 1997.
- [34] C.K Toh. Associativity-based routing for ad hoc mobile networks. 4(2) :103–139, March 1997.
- [35] R. Dube, C.D. Rais, Kuang-Yeh Wang, and S.K. Tripathi. Signal stability-based adaptive routing (ssa) for ad hoc mobile networks. *Personal Communications, IEEE*, 4(1) :36–45, Feb 1997.

- [36] Y.B. Ko and N.H. Vaidya. Location-aided routing (LAR) in mobile ad hoc networks. *Wirel. Netw.*, 6(4) :307–321, July 2000.
- [37] N.H. Saeed, M.F. Abbod, and H.S. Al-Raweshidy. Manet routing protocols taxonomy. In *International Conference on Future Communication Networks (ICFCN)*, pages 123–128, April 2012.
- [38] C. Adjih, P. Minet, T. Plesse, A. Laouiti, A. Plakoo, M. Badel, P. Mühlethaler, P. Jacquet, and J. Lecomte. Experiments with OLSR Routing in a MANET (RTO-MP-IST-054). In *Meeting Proceedings RTO-MP-IST-054*, pages 23–1 – 23–24, 2006.
- [39] V.C.M. Borges, M. Curado, and E. Monteiro. Cross-layer routing metrics for mesh networks : Current status and research directions. *Comput. Commun.*, 34(6) :681–703, May 2011.
- [40] M.A. Ngadi, S. Ali, A.H. Abdullah, and R.H. Khokhar. A taxonomy of cross layer routing metrics for wireless mesh networks. volume 2012, page 177, 2012.
- [41] F. Entezami and C. Politis. Routing protocol metrics for wireless mesh networks. In *In : Wireless World Research Forum Meeting 30 (Unpublished)*, 2013.
- [42] G. Parissidis, M. Karaliopoulos, R. Baumann, T. Spyropoulos, and B. Plattner. Routing metrics for wireless mesh networks. pages 199–230, jan 2009.
- [43] J.L. Sobrinho. Algebra and algorithms for qos path computation and hop-by-hop routing in the internet. *IEEE/ACM Trans. Netw.*, 10(4) :541–550, August 2002.
- [44] J.L. Sobrinho. Network routing with path vector protocols : Theory and applications. In *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, SIGCOMM '03*, pages 49–60, New York, NY, USA, 2003. ACM.
- [45] Hongwei Z., Anish A., and Prasun S. Learn on the fly : Data-driven link estimation and routing in sensor network backbones. In *INFOCOM*. IEEE, 2006.
- [46] H. Lundgren, E. Nordström, and C.F Tschudin. Coping with communication gray zones in IEEE 802.11b based ad hoc networks. In *The Fifth International Workshop on Wireless Mobile Multimedia, WOWMOM 2002, Atlanta, Georgia, 28 September 2002, Proceedings*, pages 49–55, 2002.
- [47] S.S.N. Rao, Y.K.S. Krishna, and K.N. Rao. Performance evaluation of routing protocols in wireless mesh networks. *International Journal of Computer Applications*, 68(7) :20–25, April 2013.
- [48] <http://www.ietf.org/>.

- [49] D.B. Johnson, D.A. Maltz, and Y.C. Hu. The dynamic source routing protocol for mobile ad hoc networks (DSR). 2003.
- [50] N. Javaid, A. Bibi, and K. Djouani. Interference and bandwidth adjusted (ETX) in wireless multi-hop networks. In *IEEE GLOBECOM Workshops (GC Wkshps)*, pages 1638–1643, 2010.
- [51] W. Jiang, S. Liu, Y. Zhu, and Z. Zhang. Optimizing routing metrics for large-scale multi-radio mesh networks. In *International Conference on Wireless Communications, Networking and Mobile Computing. WiCom 2007.*, pages 1550–1553, Sept 2007.
- [52] M.E.M. Campista, P.M. Esposito, I.M. Moraes, L.H.M.K. Costa, O.C.M.B. Duarte, D.G. Passos, C.V.N. de Albuquerque, D.C.M. Saade, and M.G. Rubinstein. Routing metrics and protocols for wireless mesh networks. *IEEE Network*, 22(1) :6–12, Jan 2008.
- [53] R. Draves, J. Padhye, and B. Zill. Comparison of routing metrics for static multi-hop wireless networks. *SIGCOMM Comput. Commun. Rev.*, 34(4) :133–144, 2004.
- [54] R.J Punnoose, P.V Nikitin, J. Broch, and D.D. Stancil. Optimizing wireless network protocols using real-time predictive propagation modeling. In *Radio and Wireless Conference (RAWCON)*, 1999.
- [55] T. Goff, N. Abu-Ghazaleh, D. Phatak, and R. Kahvecioglu. Preemptive routing in ad hoc networks. volume 63, pages 123–140, Orlando, FL, USA, February 2003. Academic Press, Inc.
- [56] R. Dube, C.D. Rais, Kuang-Yeh Wang, and S.K. Tripathi. Signal stability-based adaptive routing (ssa) for ad hoc mobile networks. *Personal Communications, IEEE*, 4(1) :36–45, Feb 1997.
- [57] U. Ashraf, S. Abdellatif, and G. Juanole. An Interference and Link-Quality Aware Routing Metric for Wireless Mesh Networks. In *68th IEEE Vehicular Technology Conference. VTC 2008-Fall.*, pages 1–5, Sept 2008.
- [58] B. Awerbuch, D. Holmer, and H. Rubens. The medium time metric : High throughput route selection in multi-rate ad hoc wireless networks. *Mob. Netw. Appl.*, 11(2) :253–266, April 2006.
- [59] K. Junhyung, Y. Jangkyu, Y. Mahnsuk, C. Keuchul, L. Honggil, and H. Kijun. A routing metric based on available bandwidth in wireless mesh networks. In *The 12th International Conference on Advanced Communication Technology (ICACT)*, 2010, volume 1, pages 844–849, Feb 2010.

- [60] I. Ullah, K. Sattar, Z.U. Qamar, W. Sami, and A. Ali. Transmissions failures and load-balanced routing metric for wireless mesh networks. In *High Capacity Optical Networks and Enabling Technologies (HONET), 2011*, pages 159–163, Dec 2011.
- [61] L.T. Nguyen, R. Beuran, and Y. Shinoda. An interference and load aware routing metric for Wireless Mesh Networks. *Int. J. Ad Hoc Ubiquitous Comput.*, 7(1) :25–37, December 2011.
- [62] V. Borges, D. Pereira, M. Curado, and E. Monteiro. Routing Metric for Interference and Channel Diversity in Multi-Radio Wireless Mesh Networks. In *Ad-Hoc, Mobile and Wireless Networks*, volume 5793, pages 55–68. 2009.
- [63] P. Bahl, A. Adya, J. Padhye, and A. Walman. Reconsidering wireless systems with multiple radios. *SIGCOMM Comput. Commun. Rev.*, 34(5) :39–46, October 2004.
- [64] R. Draves, J. Padhye, and B. Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking, MobiCom '04*, pages 114–128, New York, NY, USA, 2004. ACM.
- [65] P. Abhisek, B. Paritosh, and P.M. Santi. Designing interference based routing metric in wireless mesh network and comparison according to various tunable parameters. *International Journal of Information Sciences Letters*, 3 :21–27, 2014.
- [66] Y. Yang, J. Wang, and R.H Kravets. Designing routing metrics for mesh networks. In *IEEE WiMesh*, 2005.
- [67] P. Kyasanur and N.H. Vaidya. Routing and link-layer protocols for multi-channel multi-interface ad hoc wireless networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, 10(1) : 31–43, January 2006.
- [68] P. Subramanian, M.M. Buddhikot, and S. Miller. Interference aware routing in multi-radio wireless mesh networks. In *In Proc. of IEEE Workshop on Wireless Mesh Networks (WiMesh)*, pages 55–63, 2006.
- [69] C.E. Koksal and H. Balakrishnan. Quality-aware routing metrics for time-varying wireless mesh networks. *Selected Areas in Communications, IEEE Journal on*, 24(11) : 1984–1994, Nov 2006.
- [70] D. Cunha, O.C.M.B. Duarte, and G. Pujolle. An Enhanced Routing Metric for Fading Wireless Channels. In *IEEE Wireless Communications and Networking Conference. WCNC'08.*, pages 2723–2728, March 2008.
- [71] N. Javaid, A. Javaid, I.A. Khan, and K. Djouani. Performance study of ETX based wireless routing metrics. In *2nd IEEE International Conference on Computer. Control & Communication*, 2009.

- [72] B. Qi and S. Wu. A bottleneck aware routing metric for wireless mesh networks. In Saad Biaz and Shaoen Wu, editors, *SpringSim (ANSS)*, pages 135–141. SCS/ACM, 2011.
- [73] B. Qi, F. Shen, and S. Raza. ibatd : A new routing metric for multi-radio wireless mesh networks. In *Information Technology : New Generations (ITNG), 2012 Ninth International Conference on*, pages 502–507, April 2012.
- [74] S. Biaz and B. Qi. "iETT : A quality routing metric for multi-rate multi-hop networks". In *IEEE Wireless Communications and Networking Conference, WCNC'08*, pages 2729–2734, 2008.
- [75] S. Biaz and Y. Qi, B.and Ji. Improving Expected Transmission Time Metric in Multi-Rate Multi-Hop Networks. In *5th IEEE Consumer Communications and Networking Conference. CCNC'08*, pages 533–537, Jan 2008.
- [76] J. Jangeun, P. Pushkin, and Mihail L.S. Theoretical maximum throughput of IEEE 802.11 and its applications. In *Proceedings of the IEEE International Symposium on Network Computing and Applications*, pages 249–257, 2003.
- [77] S. Agarwal, A. Ahuja, J.P. Singh, and R. Shorey. Route-lifetime assessment based routing (RABR) protocol for mobile ad-hoc networks. In *IEEE International Conference on Communications. ICC 2000.*, volume 3, pages 1697–1701 vol.3, 2000.
- [78] K. Paul, S. Bandyopadhyay, A. Mukherjee, and D. Saha. Communication-aware mobile hosts in ad-hoc wireless network. In *Personal Wireless Communication, 1999 IEEE International Conference on*, pages 83–87, 1999.
- [79] A.B. McDonald and T. Znati. Predicting node proximity in ad-hoc networks : a least overhead adaptive model for selecting stable routes. In *Proceedings of the 1st ACM, MobiHoc 2000, Boston, Massachusetts, USA*, pages 29–33, 2000.
- [80] A.B. McDonald and T. Znati. A path availability model for wireless ad-hoc networks. In *Proc. IEEE Wireless Communications and Networking Conference, WCNC'99*, pages 35–40, 1999.
- [81] Ch. Yawut, B. Paillassa, and R. Dhaou. Mobility metrics evaluation for self-adaptive protocols. *Journal of Networks*, 3(1) :53–64, janvier 2008.
- [82] J. Tsumochi, K. Masayama, H. Uehara, and M. Yokoyama. Impact of mobility metric on routing protocols for mobile ad hoc networks. In *IEEE Pacific Rim Conference on Communications, Computers and signal Processing, 2003. PACRIM.*, volume 1, pages 322–325 vol.1, Aug 2003.

- [83] M. Ghassemian, V. Friderikos, and A.H Aghvami. *A novel algorithm for supervisory control by monitoring mobility and traffic in wireless ad hoc networks*. Wireless World Research Forum 12 meeting, 2004.
- [84] M. Ghassemian, M. Mostafavi, V. Friderikos, and A.H Aghvami. On mobility applied for ad hoc network protocol evaluation. In *The 7th IFIP International Conference on Mobile and Wireless Communications Networks (MWCN 2005)*, Septembre 2005.
- [85] F. Bai, Narayanan Sadagopan, and A. Helmy. Important : a framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks. In *Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. INFOCOM 2003*, volume 2, pages 825–835 vol.2, March 2003.
- [86] J. Boleng, W. Navidi, and T. Camp. Metrics to enable adaptive protocols for mobile ad hoc networks. In *Proceedings of the International Conference on Wireless Networks (ICWN '02)*, page 293–298, June 2002.
- [87] E. Hyttiä and J. Virtamo. Random Waypoint Mobility Model in Cellular Networks. *Wirel. Netw.*, 13(2) :177–188, April 2007.
- [88] M. Gerharz, C. de Waal, M. Frank, and P. Martini. Link stability in mobile wireless ad hoc networks. In *27th Annual IEEE Conference on Local Computer Networks. LCN 2002.*, pages 30–39, Nov 2002.
- [89] M. Ghassemian, M. Mostafavi, V. Friderikos, and A.H. Aghvami. On mobility applied for ad hoc network protocol evaluation. In *The 7th IFIP International Conference on Mobile and Wireless Communications Networks (MWCN 2005)*, Septembre 2005.
- [90] B.J. Kwak, N.O. Song, and L.E. Miller. A standard measure of mobility for evaluating mobile ad-hoc network performance. *IEICE Transactions on Communications*, E86-B : 3236–3243, November 2003.
- [91] <http://www.nsnam.org/>.
- [92] T. Clausen, C. Dearlove, and J. Dean. Mobile Ad Hoc Network (MANET) Neighborhood Discovery Protocol (NHDP), RFC6130, apr 2011. URL <http://www.ietf.org/rfc/rfc6130.txt>.
- [93] Webcasts University of Agder White Papers and Case Studies TechRepublic. Load-balancing in multi-homed OLSR networks using airtime metric and radio-aware path selection : Implementation and testing. URL <http://whitepapers.techrepublic.com.com/abstract.aspx?docid=375136>.
- [94] IEEE 802.11 WG TGs. Draft amendment to standard IEEE 802.11TM :ESS mesh networking. P802.11sTM /D1.08, Jan 2008.

- [95] T. Clausen, C. Dearlove, P. Jacquet, and U. Herberg. The optimized link state routing protocol version 2, RFC7181, apr 2014. URL <http://www.ietf.org/rfc/rfc7181.txt>.
- [96] S. Keshav. A control-theoretic approach to flow control. *SIGCOMM Comput. Commun. Rev.*, 21(4) :3–15, August 1991.
- [97] A. Bashyam, A. Jeevan, S.K. Tripuramalla, and S.B. Sanka. Effective path identification protocol for wireless mesh networks. *International Journal on Communication*, 1(3) :3, December 2010.
- [98] <http://wireless.kernel.org/en/developers/Documentation/mac80211/RateControl/minstrel>.
- [99] E. Weingartner, H. vom Lehn, and K. Wehrle. A performance comparison of recent network simulators. In *IEEE International Conference on Communications. ICC '09.*, pages 1–5, June 2009.
- [100] A. Ouni, H. Rivano, and F. Valois. Capacité de réseaux maillés sans fil : éléments déterminants et caractères insensibles. In *11eme Journées Doctorales en Informatique et Réseaux (JDIR)*, Sophia Antipolis, France, March 2010.
- [101] S.L. Menezes. *Optimization of Handovers in Present and Future Mobile Communication Networks*. ProQuest, UMI Dissertation Publishing, 2011.
- [102] J.M. Francois. Performing and making use of mobility prediction. *Doctor thesis, University of Lyte*, 2007.
- [103] G. Yavas, D. Katsaros, Ö.ü. Ulusoy, and Y. Manolopoulos. A data mining approach for location prediction in mobile environments. *Data Knowl. Eng.*, 54(2) :121–146, August 2005.
- [104] A.E. Bergh and N. Ventura. Prediction assisted fast handovers for mobile ipv6. *IEEE MILCOM 2006 Unclassified Technical Sessions, Washington D.C.*, 2006.
- [105] Zhang D.W. and Y. Yao. A Predictive Handoff Approach for Mobile IP. In *International Conference on Wireless and Mobile Communications. ICWMC '06.*, pages 78–78, July 2006.
- [106] S.S. Ali, Kamalrulnizam A.B., Z.G. Kayhan, and A.J Gonzalez. Mobility and Signal Strength- Aware Handover Decision in Mobile IPv6 based Wireless LAN. *Proceeding of the International MultiConference of Engineers and Computer Scientists, IMECS'2011, Hong Kong*, 1, march .
- [107] X. Chen, F. Mériaux, and S. Valentin. Predicting a User's Next Cell With Supervised Learning Based on Channel States. In *The 14th IEEE International Workshop on Signal Processing Advances for Wireless Communications (SPAWC)*, pages 1–5, Darmstadt, Germany, June 2013.



- [108] J. Capka and R. Boutaba. Mobility Prediction in Wireless Networks Using Neural Networks. In Vicente, John and Hutchison, David, editor, *Management of Multimedia Networks and Services*, volume 3271 of *Lecture Notes in Computer Science*, pages 320–333. Springer Berlin Heidelberg, 2004.
- [109] S.M. Mousavi, H.R. Rabiee, M. Moshref, and A. Dabirmoghaddam. Model Based Adaptive Mobility Prediction in Mobile Ad-Hoc Networks. In *International Conference on Wireless Communications, Networking and Mobile Computing. WiCom 2007.*, pages 1713–1716, Sept 2007.
- [110] X. Li, N. Mitton, and D. Simplot-Ryl. Mobility Prediction Based Neighborhood Discovery in Mobile Ad Hoc Networks. In *Networking 2011*, pages 147–159, Valencia, Spain, May 2011.
- [111] S. Naimi, A. Busson, V. Veque, Larbi. Ben Hadj Slama, and R. Bouallegue. Mobility management in ad hoc networks using routing metrics. In *The fourth IEEE International Conference on Communications and Networking (ComNet)*, pages 1–6, March 2014.
- [112] A. Duel-Hallen, S. Hu, and S. Hallen. Long range prediction of fading signals : Enabling adaptive transmission for mobile radio channels. *IEEE Signal Processing Magazine*, 17 : 62–75, 2000.
- [113] X. Long and B. Sikdar. A real-time algorithm for long range signal strength prediction in wireless networks. In *IEEE Wireless Communications and Networking Conference. WCNC'08.*, pages 1120–1125, 2008.
- [114] V. Erceg and al. An empirically based path loss model for wireless channels in suburban environments. *IEEE Journal on Selected Areas in Communications*, 17(7), Jul 1999.
- [115] I. Stepanov and K. Rothermel. On the Impact of a More Realistic Physical Layer on MANET Simulations Results. volume 6, pages 61–78. Elsevier Science Publishers B. V., January 2008.
- [116] H. Badis and K. Al Agha. Optimal Path Selection in Ad Hoc Networks Based on Multiple Metrics : Bandwidth and Delay. In *Advances in Wireless Networks and Mobile Computing*, pages 35–60. Springer US, 2008.
- [117] J. Yi, A. Adnane, S. David, and B. Parrein. Multipath Optimized Link State Routing for Mobile Ad Hoc Networks. *Ad Hoc Netw.*, 9(1) :28–47, January 2011.
- [118] E.A. Panaousis, G. Drew, G.P. Millar, T.A. Ramrekha, and C. Politis. A Testbed Implementation for Securing OLSR in Mobile Ad hoc Networks. *International Journal of Network Security Its Applications*, 2(4), October 2010.

- [119] M. Hiyama, E. Kulla, M. Ikeda, L. Barolli, and M. Takizawa. Performance Comparison of OLSR with ETX\_Float and ETX\_ff in a MANET Testbed. In *CISIS*, pages 8–14. IEEE, 2013.
- [120] <http://www.olsr.org>. [Online].
- [121] C. Houaidia, A. van den Bossche, H. Idoudi, T. Val, and L.A. Saidane. Experimental performance analysis of routing metrics in wireless mesh networks. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2013 9th International*, pages 1011–1016, July 2013.
- [122] <http://www.ubuntu-fr.org/telechargement>. [Online].
- [123] <http://www.cjoint.com/15mi/EECpznTLqbQ.htm>. [Online].
- [124] [http://opensource.telkomspeedy.com/wiki/index.php/OLSR\\_-\\_di\\_Ubuntu](http://opensource.telkomspeedy.com/wiki/index.php/OLSR_-_di_Ubuntu). [Online].
- [125] <https://iperf.fr>. [Online].

---

**Résumé :** Avec le succès des communications sans fil, il devient possible d'accéder au réseau partout et à tout moment sans avoir recours à connecter physiquement les appareils communicants à une infrastructure. Les noeuds (ordinateurs portables, smartphones, etc) peuvent analyser les différents canaux radio afin de pouvoir s'associer à un réseau sans fil disponible (station de base, point d'accès, etc.). Un avantage indéniable de ses technologies sans fil est la possibilité d'être mobile tout en restant connecté. Cependant, la mobilité est une tâche difficile à gérer car elle doit être abordée à différentes couches pour être transparente aux utilisateurs. Dans les MANET (Mobile Ad hoc Network), les protocoles de routage utilisent des métriques pour sélectionner les meilleures routes. Les métriques peuvent refléter la qualité de la liaison sans fil et aider à gérer la mobilité. Mais, un retard important entre l'estimation des métriques et leur inclusion dans le processus de routage rend cette approche inefficace.

Les travaux de cette thèse s'intéressent à la proposition de nouvelles méthodes de calcul des métriques de routage pour gérer le problème de la mobilité dans les réseaux ad hoc. Les nouvelles métriques doivent refléter la qualité du lien et être sensibles à la mobilité en même temps. Nous considérons les métriques classiques, en particulier ETX (Expected Transmission Count) et ETT (Expected Transmission Time). Nous introduisons de nouvelles méthodes pour anticiper les valeurs de ces métriques en utilisant des algorithmes de prédiction. Nous utilisons une approche Cross Layer, qui permet l'utilisation conjointe de l'information à partir des couches 1, 2 et 3. La validation de nouvelles méthodes de calcul des métriques de routage nécessite une évaluation au travers d'un véritable banc d'essai. Nous avons donc également mis en oeuvre les nouvelles métriques de routage dans un testbed afin d'évaluer et de comparer leurs performances avec les métriques classiques.

**Mots clés :** ETX, métrique, Ad Hoc, réseau sans fil, mobilité, anticipation

---